

00010010011001001001001001000100100100010001000010010010001000

OUTRAS APOSTILAS EM:
www.projetoderedes.com.br

Gerência de Redes

baseado em

Software Livre

Emerson Silva de Souza
Emerson.network@hotmail.com.br

00010010011001001001001001000100100100010001000010010010001000

Nota de Copyleft

Copyleft © 2007 – Emerson Silva de Souza

É permitida a cópia, distribuição e modificação total ou parcial deste documento sob os termos da Licença Livre para Documentação GNU (GNU FDL), versão 1.2 ou qualquer versão posterior publicada pela Free Software Foundation.

Introdução

Este trabalho foi motivado por uma grande falta de gerenciamento dos recursos da rede, e de se trabalhar sempre na forma reativa (apaga fogo) e não na forma pró-ativa, pois precisamos identificar os problemas logo que eles ocorrem para também resolver com velocidade melhor.

Gerenciar o ambiente de rede é uma das preocupações mais constantes entre empresas e organizações. Dispor de ferramentas que façam esse controle é fundamental para facilitar o trabalho e identificar imediatamente algum tipo de erro providenciando assim uma ação efetiva. Ela destaca também a importância da adoção desses mecanismos de segurança. "A rede de uma empresa está em ameaça constante por isso sempre priorizamos a segurança com ações que representem pró-atividade com relação a possíveis problemas.

Um dos aspectos destacados nesse tipo de solução é opção por controle através de gráficos e relatórios, além de alertas pelos quais o administrador pode ter a opção de ser avisado se acontecer qualquer instabilidade na rede. Proporcionando um acompanhamento em tempo real dos acontecimentos.

A Secretaria Municipal de Ciência e Tecnologia da Informação como principal centro de tecnologia da Informação das Secretarias Municipais, tem por obrigação oferecer aos seus clientes uma solução completa em monitoramento focadas em maior segurança no desenvolvimento das aplicações, mantendo a rede.

Atualmente estão sendo utilizadas uma série de programas para esse controle com sucesso . O sistema de monitoramento que implementamos é composto pelo **NAGIOS**, **CACTI+WEATHEMAP** e **SMOKEPING**, além de vários softwares de segurança mas não é o escopo deste artigo, pois abordaremos somente sobre as ferramentas destacadas acima.

Sumário

1. Gerência e Monitoramento de Redes	5
1.1 Independência de fornecedor:	5
1.2 Utilização de padrões abertos:	5
1.3 Baixo custo de implementação	5
1.4 Bons softwares	5
1.5 Gerência de Falhas	6
1.6 Gerência de Desempenho	6
1.7 Configuração Centralizada	6
1.8 Modelos de Gerenciamento de Redes	7
1.9 Requisitos do Sistema	7
2. Instalando e configurando o Nagios	8
2.1 Introdução	8
2.2 Iniciando a Instalação	8
2.3 Instalando os Plugins do Nagios	9
2.4 Configurando o Apache	10
2.5 Configurando o Nagios	10
2.6 Descrição dos arquivos de configuração	11
2.7 Verificando as configurações	15
2.8 Iniciando o Nagios	15
3. Instalando e configurando o Cacti	16
3.1 Introdução	16
3.2 Iniciando a Instalação	16
3.3 Instalando e configurando o Mysql	16
3.4 Apache e PHP	17
3.5 Testando a configuração do PHP	18
3.6 Instalando o RRDTOOL	18
3.7 Instalando o NET-SNMP	18
3.8 Instalando o Cacti	18
3.9 Finalizando as configurações	19
4. Configurando a Arquitetura de Plugins	20
4.1 Instalando	20
4.2 Usando pacotes pré-compactados	20
4.4 Configuração	21
4.5 Instalando os plugins	21
4.6 Ativando o plugin	21
5. Integrando o PHP-Weathermap com o cacti	23
5.1 Introdução	23
5.2 Exigências	23
5.3 Tipos de Instalação	23
5.4 Iniciando a instalação	24
5.5 Instalando como plugin do cacti	24
5.6 Configurando o editor	25
5.7 Princípios Básicos	26
5.8 Ajustes globais	27
5.9 Uma Configuração de exemplo	28
6. Instalando e Configurando o Smokeping	32
6.1 Introdução	32
6.2 Alarmes expertos	32
6.3 Sustentação dinâmica do IP	32
6.4 Fundo técnico	32
6.5 Pré-requisitos	33
6.6 Instalação	35
6.7 Requerimentos	35
6.8 Instalação do fping	35
6.9 Instalação do cgi	35
6.11 Instalação Smokeping	36
6.12 Configurando	36
6.13 SmokePing Configuração	37
7. Considerações Finais	42
8. Referencias Bibliográficas	43

1. Gerência e Monitoramento de Redes

Para mantermos uma rede estável precisamos atentar para as principais características que por muitas vezes esquecemos devidos aos vícios que adquirimos todos os dias, devemos seguir os procedimentos básicos e necessários para gerenciar e monitorar uma rede de computadores fazendo acompanhamento dos eventos de uma rede, a fim de diagnosticar problemas e determinar quando e quais procedimentos de contingência devem ser aplicados, bem como obter estatísticas para administração e otimizações de desempenho, que são elas:

1.1 Independência de fornecedor:

Empresas não duram para sempre. Algumas soluções tornam-se obsoletas e não encontram sustentabilidade por dependerem de um só fornecedor, é também importante obter um hardware que siga padrões estabelecidos, e que seja suportado por diferentes sistemas operacionais.

1.2 Utilização de padrões abertos:

Alguém montaria sua rede corporativa implementando uma solução não compatível com o TCP/IP? a exemplo do SNMP (Simple Network Management Protocol), que é um protocolo para gerência de redes bastante popular, com especificação aberta e em constante evolução. Mesmo assim alguns fabricantes empurram seus protocolos de gerência, e algumas pessoas pagam caro (nos dois sentidos) por isso. Alguns fabricantes fazem pior. Empacotam protocolos estabelecidos num rótulo com sua assinatura.

1.3 Baixo custo de implementação

Montar uma rede não é barato. A infraestrutura custa caro, a implantação custa caro e o suporte custa caro. O software pode custar caro também, mas essa é uma escolha sua.

1.4 Bons softwares

Bons softwares de gerência e monitoramento devem prever situações indesejáveis na rede, e quando possível tomar as ações preventivas pelo administrador. Uma interface clara e a centralização das informações são também características importantíssimas para a gerência da rede. A independência de plataforma é essencial. É muito difícil encontrar uma rede homogênea nos quesitos software e hardware. Aí que geralmente somos ludibriados com programas que se dizem multi-plataformas. É muito raro um software proprietário atender a esse requisito. Geralmente encontramos uma solução “multi-plataforma” porque que roda em Windows{XP,2000,NT,etc} e no máximo num redhat7.2 ou FreeBSD4.3.

Vamos focar aqui nas três preocupações básicas que um administrador de redes deve ter:

1.5 Gerência de Falhas

Aqui temos as tarefas de (1) Identificar a falha, (2) Isolar a falha e (3) Corrigir a falha. É uma das tarefas mais importantes para deixar uma rede com um bom nível de confiabilidade.

É também uma tarefa complexa, necessita de ferramentas bastante escaláveis e de preferência que automatizem ao máximo o processo, desde a identificação até a correção de uma falha.

1.6 Gerência de Desempenho

Muito importante para (1) evitar o mau uso da rede e (2) prover uma capacidade de planejamento. Gerenciar o desempenho dos recursos de uma rede é muito importante para identificar quais deles devem ser revisto a propósito de upgrades, ou mesmo de diminuição de custos (largura de banda desnecessária, por exemplo). Essencial também para detectar atividades ilícitas na rede, como por exemplo a detecção de utilização dos recursos durante períodos em que não deveria existir.

1.7 Configuração Centralizada

A centralização das informações, tal como seu correlacionamento é uma característica muito importante para a administração de uma rede. Uma interface simples de se manipular é essencial quando se trata de dados complexos e provenientes de variadas localidades.

Após a consolidação da WEB ficou muito mais fácil produzir ambientes de gerência (NMS) remotos multi-plataforma e amigáveis. Criou-se então uma forte independência de GUIs proprietários, possibilitando a gerência remota em qualquer sistema conectado à rede.

Praticamente todas as infra-estruturas de redes requerem atuação operacional constante, em suas várias aplicações:

- Backbones acadêmicos e comerciais
- Redes metropolitanas
- Provedores de acesso Internet (ISPs)
- Pontos de Troca de Tráfego (PTT, NAPs, IXPs)
- Datacenters
- Redes corporativas

Assim, cada vez mais:

- São exigidos maiores níveis de qualidade dos serviços,

- Aumenta a criticidade da operação.
- É requerido o uso de boas e confiáveis ferramentas de apoio.

1.8 Modelos de Gerenciamento de Redes

FCAPS – acrônimo referente às áreas funcionais de gerenciamento definidas no modelo ISO de gerenciamento de redes **ITIL** – Biblioteca de Infra-estrutura de T.I. (Information Technology Infrastructure Library)

(S)ecurity)	Segurança
(P)erformance)	Desempenho
(A)ccounting)	Contabilização
(C)onfiguration)	Configuração
(F)ault)	Falhas

Vamos agora apresentar algumas ferramentas livres que cumprem com os requisitos básicos para gerência e monitoramento de recursos de redes.

1.9 Requisitos do Sistema

É necessário que se tenha um servidor WEB instalado e devidamente configurado. Estaremos tratando do Apache neste documento. É preciso também ter instalado a biblioteca GD versão 1.6.3 ou posterior para a geração dos gráficos e outras bibliotecas que serão elencadas ao decorrer deste artigo.

Todas as aplicações foram instaladas na plataforma linux em slackware versão 10.1

2. Instalando e configurando o Nagios

2.1 Introdução

Nagios é um programa open source de monitoramento de redes que verifica constantemente a disponibilidade do serviço, local ou remoto e avisa por meio de email ou celular sobre o problema ocorrido.

É possível obter relatórios de disponibilidade e configurar ações corretivas para os problemas ocorridos na rede. O Nagios tem uma interface que permite monitoramento via WAP e Browser (Netscape, Mozilla, Galeon...), possui configuração de alertas, o que permite tomar ações corretivas para um erro eminente.

2.2 Iniciando a Instalação

Faça o download dos arquivos no endereço abaixo:

<http://www.nagios.org/download/>

Descompacte o arquivo com o código fonte, use o comando:

```
tar -xvzpf nagios-2.6.tar.gz
```

Crie o diretório de instalação:

```
mkdir -p /usr/local/nagios
```

Adicione o usuário e grupo nagios:

```
groupadd nagios
```

```
useradd -g nagios -d /usr/local/nagios -s /bin/false nagios
```

NOTA: Verifique se você já tem instalado o componente abaixo, caso ainda não tenha favor instalar:

- **GD-Library**

Caso esteja faltando você não conseguirá abrir o statusmap, que é o mapa principal do nagios.

A compilação será básica, sem suporte a banco de dados, para isso foi usado os comandos:

```
./configure --prefix=/usr/local/nagios --with-nagios-user=nagios  
--with-nagios-grp=nagios --with-cgiurl=/nagios/cgi-bin --with-htmurl=/nagios
```

Parâmetros para compilação:


```
make all -s ( criação de binários )  
make install -s ( instalação de binários )  
make install-init -s ( instalação de script de inicialização )  
make install-commandmode -s ( Configuração de permissões )  
make install-config -s ( instalação de exemplos de configuração no diretório  
/usr/local/nagios/etc )
```

Após a instalação serão criados no diretório do nagios (/usr/local/nagios) previamente criado, os seguintes subdiretórios:

- bin/ (Arquivos binários do Nagios)
- etc/ (Diretório de Configuração)
- sbin/ (CGI's)
- share/ (Arquivos HTML)
- var/ (Diretório de logs)

2.3 Instalando os Plugins do Nagios

Descompacte o arquivo com o código fonte, use o comando:

```
tar -xvzf nagios-plugins-1.4.6.tar.gz
```

NOTA: Verifique se não está faltando algum componente para os plugins.
Geralmente falta o componente abaixo caso você ainda não tem instalado:

- **Radiusclient**

faça o download antes de continuar com a compilação .

Para compilar use o seguinte parâmetro:

```
./configure --prefix=/usr/local/nagios --with-nagios-usr=nagios  
--with-nagios-grp=nagios
```

```
make all -s ( criação de binários )  
make install -s ( instalação de binários )
```

Após a instalação será criado um novo diretório dentro do /usr/local/nagios conforme abaixo:

/usr/local/nagios

– libexec/ (Arquivos de Plugins do Nagios)

2.4 Configurando o Apache

Para acessar o Nagios via web, é necessário adicionar a seguinte configuração no httpd.conf: (/etc/apache).

```
ScriptAlias /nagios/cgi-bin /usr/local/nagios/sbin/
```

```
<Directory "/usr/local/nagios/sbin/">
    AllowOverride AuthConfig
    Options ExecCGI
    Order allow,deny
    Allow from all
    AuthName "Acesso ao Nagios"
    AuthType Basic
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    require valid-user
</Directory>
```

```
Alias /nagios /usr/local/nagios/share/
```

```
<Directory "/usr/local/nagios/sbin/">
    Options None
    AllowOverride AuthConfig
    Order allow,deny
    Allow from all
    AuthName "\"Acesso Nagios\""
    AuthType Basic
    AuthUserFile /usr/local/nagios/etc/htpasswd.users
    require valid-user
</Directory>
```

Criar o arquivo htpasswd.users que vai conter os nomes e as senhas dos usuários que vão ter acesso ao Nagios.

Para criar a senha utilize o comando abaixo:

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users <USUARIO>
```

Pronto, agora o nagios está protegido com senha, vamos para o próximo passo.

2.5 Configurando o Nagios

Os arquivos de configuração estão dentro do diretório /usr/local/nagios/etc/. Todos os arquivos vem com o .sample na frente, que são exemplos, pode-se tirar os .sample dos arquivos e configura-los ou fazer uma cópia deles, que é o mais aconselhável. Mas para funcionarem não podem continuar com o .sample na frente.

2.6 Descrição dos arquivos de configuração

nagios.cfg :

Arquivo com as configurações principais. O padrão é bem completo, somente altere os parâmetros:

check_external_commands=0 para check_external_commands=1
date_format=us para date_format=euro

cgi.cfg :

Arquivo de configuração dos programas CGIs localizados na pasta sbin.

Nele ficam as configurações de utilização de arquivos cgi pelo Nagios. Devem ser configurados os parâmetros de autorização de utilização da interface Web. Altere os campos para o nome do usuário cadastros no arquivo /usr/local/nagios/etc/htpasswd.users e assim terão acesso ao sistema, os parâmetros são:

```
authorized_for_system_information=<usuario>
authorized_for_configuration_information=<usuario>
authorized_for_system_commands=<usuario>
authorized_for_all_services=<usuario>
authorized_for_all_hosts=<usuario>
authorized_for_all_service_commands=<usuario>
authorized_for_all_host_commands=<usuario>
```

hosts.cfg :

Arquivo contendo informações dos hosts.

Armazenas as informações sobre os dispositivos de rede (computador, switch, impressora, roteador).

Um exemplo de hosts.cfg com apenas uma máquina:

```
define host {
    use generic-host      ; Name of host template to use
    host_name              computador.dominio.com.br
    alias                  computador
    address                192.168.0.1
    check_command          check-host-alive
    max_check_attempts    10
    notification_interval  120
    notification_period    24x7
    notification_options   d,u,r
}
```

host_name: é o nome do computador na rede.

Alias: é um apelido amigável para o computador.

Address: é o endereço IP da máquina.

check_command: é o comando a ser executado, está definido em checkcommands.cfg.

max_check_attempts: é o número máximo de vezes que o programa irá checar pelo serviço até reportar um erro.

notification_interval: de quantos em quantos segundos será enviado uma notificação para o usuário responsável pelo serviço.

notification_period: qual o horário que o serviço ficará ativo.

notification_options: quando notificar um erro, d-down,u-unrecheable,r-recovery.

hostgroups.cfg :

Arquivo contendo informações dos hosts por grupos.

Armazena informações de grupos de equipamentos (computadores, switch, impressoras) existentes em sua rede.

A configuração de hostgroups.cfg torna-se necessário para que se organize os dispositivos de rede e torne a leitura mais fácil.

Um exemplo de hostgroups.cfg:

```
define hostgroup {
    hostgroup_name    servidores
    alias             Servidores windows e linux
    contact_groups    administrador
    members           computador.dominio.com.br
}
```

contacts.cfg :

Contatos que deverão ser notificados caso ocorra algum problema.

Um exemplo de contacts.cfg:

```
define contact {
    contact_name      wberbert
    alias             Wanderson Berbert
    service_notification_period workhours
    host_notification_period  workhours
    service_notification_options c,r
    host_notification_options  d,r
    service_notification_commands notify-by-email
    host_notification_commands host-notify-by-email
    email             wberbert@sermap.com.br
}
```

contactgroups.cfg :

Contatos divididos em grupos.

Um exemplo de contactgroups.cfg:

```
define contactgroup {
    contactgroup_name    administrador
    alias                Administradores de rede
    members              wberbert
}
```

service.cfg :

Serviços que deverão ser monitorados

```
# Service definition
define service{
    use generic-service ; Name of service template to use
    host_name micro1
    service_description PING
    is_volatile 0
    check_period 24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups administrador
    notification_interval 120
    notification_period 24x7
    notification_options c,r
    check_command check_ping!100.0,20%!500.0,60%
}
```

hostextinfo.cfg:

Onde serão definidos as imagens que serão apresentadas no statusmap.

O diretório padrão das imagens é /usr/local/nagios/share/images/logos/. Para funcionar é preciso adicionar uma linha no arquivo cgi.cfg

```
xedtemplate_config_file=/usr/local/nagios/etc/hostextinfo.cfg
```

Existe outro arquivo que é o serviceextinfo.cfg que coloca imagens para os serviços, de uma olhada no arquivo de configuração ele é muito simples.

Exemplo do hostextinfo.cfg

```
# Micro1
```

```
define hostextinfo{
```

```
name micro1
icon_image linux40.png
icon_image_alt Estação 1
vrml_image linux40.png
gd2_image linux40.gd2
register 0
}

define hostextinfo{
use micro1
host_name Micro1
2d_coords 65,425
3d_coords 3.0,0.0,0.0
}
```

dependencies.cfg :

Informações de serviços que dependem de outros serviços.

timeperiods.cfg :

Informações sobre o período de monitoramento, podem ser definidos vários períodos de monitoramento diferentes.

checkcommands.cfg :

Definição dos comandos que podem ser executados pelo Nagios.

Configuração dos plugins que estão no diretório /usr/local/nagios/libexec.

resource.cfg :

Macros definidas pelos usuários.

Configuração de parâmetros de recursos. Por padrão vem definido qual o caminho dos plugins. Pode ser setadas variáveis para serem utilizadas nos arquivos CHECKCOMMANDS.CFG e MISCCOMMANDS.CFG.

misccommands.cfg:

Definição de alguns comandos, tais como envio de e-mail, envio de pager, etc

Esses são os arquivos que são legais de colocar alguns exemplos, os outros é só verificar o próprio arquivo, sendo que todos eles tem bons exemplos.

2.7 Verificando as configurações.

Para verificar se a configuração está correta use o seguinte comando:
`/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg`

Com esse comando é possível verificar o erros e os warnings, e ver em qual arquivo está o erro. O Nagios não vai iniciar se existir algum erro.

2.8 Iniciando o Nagios.

`/usr/local/nagios/bin/nagios -d /usr/local/nagios/etc/nagios.cfg`

Acesse através:

`http://localhost/nagios/`

3. Instalando e configurando o Cacti

3.1 Introdução

Ferramenta para gerar gráficos de desempenho. O programa funciona em cima de uma plataforma SNMP - Simple Network Management Protocol, que é um protocolo de gerenciamento. Ou seja, depois de configurado, o CACTI gerencia vários dados que o SNMP fornece, como: espaço em disco, utilização da CPU, tráfego de rede, entre outros.

O programa gera gráficos em tempo real o que possibilita uma ação imediata perante qualquer alteração, parecido com o NAGIOS, o software permite através da configuração de plugins a configuração de mensagens de alertas para um melhor gerenciamento e controle das ações.

3.2 Iniciando a Instalação

3.3 Instalando e configurando o Mysql

```
# installpkg /mnt/cdrom/slackware/ap/mysql-3.x.x.x.tgz
```

MySQL instalado, vamos à configuração dele:

Precisamos instalar o database:

```
# mysql_install_db
```

Precisamos setar a senha de root, mas para isso teremos que alterar um detalhe. Iniciar o mysql:

```
# mysqld_safe &
```

Vai dar um erro, é normal e necessário para gerar o sock. Agora devemos fazer o seguinte:

```
# chown -R mysql:mysql /var/lib/mysql
```

Tentaremos novamente executar o server mysql:

```
# mysqld_safe &
```

MySQL rodando, vamos para o próximo passo:

Setar a senha de root:

```
# mysqladmin -u root password (SUA SENHA AQUI)
```

Logue-se ao mysql para testar a senha

Digite no shell o comando abaixo:


```
# mysql -p
```

Vai pedir a senha: digite a sua senha;
Logado, digite exit.

Bom acabamos a parte de instalação agora vamos configurar o mysql

Digite os passos abaixo no shell do linux

```
groupadd cacti  
useradd -g cacti cactiuser
```

Logue-se ao mysql:

```
# mysql -p
```

Vai pedir a senha: digite a sua senha

```
create database cacti;  
grant all on cacti.* to root;  
grant all on cacti.* to root@localhost;  
grant all on cacti.* to cactiuser;  
grant all on cacti.* to cactiuser@localhost;  
set password for cactiuser@localhost=password('password');  
exit
```

3.4 Apache e PHP

```
# installpkg /mnt/cdrom/slackware/n/apache-1.3.x.x.x.tgz
```

```
# installpkg /mnt/cdrom/slackware/n/php-4.x.x.x.x.tgz
```

Após instalar os dois, devemos editar o arquivo /etc/apache/httpd.conf e inserir os seguintes dados:

```
# Na sessão "Dynamic Shared Object (DSO) Support":
```

```
LoadModule php4_module libexec/libphp4.so
```

```
AddModule mod_php4.c
```

```
# Na sessão DirectoryIndex, localize o IF abaixo e o edite dessa forma:
```

```
<IfModule mod_dir.c>
```

```
DirectoryIndex index.html index.php index.htm
```

```
</IfModule>
```

```
# Aproveite e insira mais este IF:
```

```
<IfModule mod_php4.c>
```

```
AddType application/x-httpd-php .php4 .php3 .phtml .php
```

```
AddType application/x-httpd-php-source .phps
```

```
</IfModule>
```

3.5 Testando a configuração do PHP

Vá na pasta /var/www/htdocs e crie um arquivo phpinfo() com este conteúdo:

```
<? phpinfo(); ?>
```

Reinicie o apache:

```
# apachectl restart
```

Abra um navegador e digite a URL: http://localhost/phpinfo.php

Se todos os passos foram seguidos sem erros, seu servidor está funcionando e você verá as informações de seu sistema :)

3.6 Instalando o RRDTOOL

Entra na pagina:

<http://linuxpackages.telecoms.bg/Slackware-10.1/Console/rrdtool/rrdtool-1.2.9-i486-2kf.tgz>

```
installpkg (nome do pacote)
```

3.7 Instalando o NET-SNMP

<http://linuxpackages.inode.at/Slackware-10.1/Console/net-snmp/net-snmp-5.2.1-i486-1gsp.tgz>

```
installpkg (nome do pacote)
```

3.8 Instalando o Cacti

<http://www.cacti.net/downloads/cacti-0.8.6j.tar.gz>

depois de baixar o pacote descompacte utilizando o comando abaixo:

```
tar -zxvf cacti-0.8.6f.tar.gz
```

renomeie a pasta utilizando o comando abaixo:

```
mv cacti-0.8.6f cacti
```

em seguida mova a pasta utilizando o comando abaixo:

```
mv cacti /var/www/htdocs/
```

Entre no diretório do cacti

```
cd /var/www/htdocs/cacti
```

```
mysql --user=root --password=password cacti < cacti.sql
```

permissões

```
chown -R cactiuser rra/ log/
```

Edit o arquivo `/var/www/htdocs/cacti/include/config.php`, conforme abaixo:

```
$database_default = "cacti";  
$database_hostname = "localhost";  
$database_username = "cactiuser";  
$database_password = "password";
```

Adicionando a linha no Crontab

Para entrar no modo de edição no crontab digite no shell:

```
crontab -e
```

adicione a seguinte linha:

```
*/5 * * * * php /var/www/htdocs/cacti/poller.php > /dev/null 2>&1
```

3.9 Finalizando as configurações

Digite no browser:

<http://localhost/cacti/index.php>

- 1- Click Next
- 2- Padrão (new install) Next
- 3- Click Finish
- 4- Login: admin
- 5- Password: admin
- 6- Click Save

Pronto, instalação concluída, agora vamos para o próximo passo, que é a instalação dos plugins que podem ser adicionados no cacti.

4. Configurando a Arquitetura de Plugins

A arquitetura de plugins foi designada para oferecer uma melhor robustez para o cacti, permitindo assim adicionar novos plugins sem a necessidade de fazer uma nova instalação.

O primeiro passo é fazer o download do Plugin Architecture.

<http://cactiusers.org/downloads/patches/>

Descompacte:

```
tar -zvcf cacti-plugin-arch.tar.gz
```

4.1 Instalando

Você pode instalar usando executando o patch ou copiando os arquivos descompactados manualmente para a pasta do cacti e sobreescrevendo-os.

4.2 Usando pacotes pré-compactados

Vá para o diretório que você extraiu a arquitetura do plugin, neste diretório você vai encontrar vários arquivos e diretórios. Procure um deles que se pareça com isso "**files-0.8.6j**" e outro parecido com "**cacti-plugin-0.8j.diff**". este arquivo pre-compactado são para o Cacti v0.8.6j,

Você irá agora copiar os dois arquivo para dentro do diretório que você instalou o cacti.

4.3 Para instalar via o patch execute os comandos abaixo:

Entre no diretório do cacti

```
# cd /var/www/htdocs/cacti
```

Execute o comando abaixo no prompt de comando:

```
# patch -p1 -N --dry-run < cacti-plugin-arch.diff
```

Verifique se aparece algum erro, se caso tudo der certo execute o comando abaixo:

```
# patch -p1 -N < cacti-plugin-arch.diff
```

Para instalar manualmente é só copiar os arquivos do diretório files-0.8.6j para dentro da pasta do cacti e sobreescrever todos.

4.4 Configuração

É necessário fazer algumas configurações depois de instalar a arquitetura de plugins.

Você precisa reeditar o `/var/www/htdocs/include/config.php`. e configurar novamente as senhas do banco.

Você precisa colocar o endereço da localização do cacti, conforme abaixo:

```
$config['url_path'] = "/cacti/";
```

Pronto, próximo passo instalando o plugins.

4.5 Instalando os plugins

Os plugins foram feitos para serem fáceis de instalar.

O primeiro passo é fazer o download do plugin da sua escolha, você pode escolher entre `.zip` e `.gzip` fica ao seu critério, você precisa criar um diretório e extrair os arquivos, bom eu vou usar o plugin monitor como exemplo deste artigo.

Inicie o download do Monitor plugin. Depois do download estar completo, o arquivo baixado parecerá como este abaixo.

monitor-0.3.zip

or

monitor-0.3.tar.gz

Descompactando

```
tar -zvcf monitor-0.3.tar.gz
```

Upload

Mova o diretório monitor, para `/var/www/htdocs/cacti/plugins/`

4.6 Ativando o plugin

Você vai precisar agora fazer uma alteração no arquivo de configuração do cacti, para poder ativar o plugin edite o arquivo abaixo.

/var/www/htdocs/cacti/include/config.php

Procure por uma linha de comando parecida com esta:

```
$plugins = array();
```

Retire os comentários

```
//$plugins[] = 'thold';
```

```
$plugins[] = 'thold';
```

Agora altere o nome, para o nome do plugin.

```
$plugins[] = 'monitor';
```

E por fim adicionar a seguinte linha no crontab

```
*/1 * * * * php /var/www/htdocs/cacti/plugins/monitor/fast_poller.php > /dev/null
```

Pronto, agora é só testar.

5. Integrando o PHP-Weathermap com o cacti

5.1 Introdução

PHP Weathermap é uma de [muitas](#) execuções da mesma idéia básica - fazer exame de dados de seus dispositivos da rede e usá-los fornecer uma vista geral single-page do estado atual da rede, Complementa uma ferramenta como [MRTG](#), [grilo](#) ou [cactis](#), que fornecem representar graficamente in-depth, e informação histórica, e pode usar dados daqueles sistemas e produzi-la em mapas.

Esta versão particular é escrita em PHP, e pode ler dados dos statistics dos dados produzidas do MRTG, arquivos de texto e dos arquivos de RRD, tais como aqueles produzidos por umas instalações mais novas de MRTG, por um cacti ou por uma outra ferramenta. Pode também gerar dados do “suporte” do HTML para as imagens do mapa, que podem incluir as folhas de prova de “DHTML” de dados e das ligações históricos em seus monitoração outras/sistema de estatísticas. Tem também um editor interativo do mapa, assim que você pode pela maior parte evitar os dados da configuração do texto, se você preferir.

5.2 Exigências

Baseado nas lições aprendidas com a versão do Perl, esta tem um jogo muito restrito das dependências - você necessitará um recentish PHP including a versão de CLI e a extensão do “gd” com sustentação do png, do TrueColour e do FreeType. Você necessita a comando-linha (*escudo do aka/ssh/telnet*) acesso ao usuário que hospedará os mapas.

Para ler limas de RRD, você necessitará a comando-linha programa do rrdtool.

Aparte do módulo do gd, estas exigências são as mesmas que para cactos, que são o sistema mais-testado do stats do sócio.

Para usar os novos plugins do cacti, você necessitará uma versão recente dos cacti, e a arquitetura Plugin combinando do cacti de [Cactiusers.org de Conner do Jimmy](#).

5.3 Tipos de Instalação

Desde que a maioria das características da integração são opcionais, há um número de maneiras instalar Weathermap.

1. Autônomo - a maneira tradicional
2. Autônomo, com editor - a opção de alguma integração com seu sistema de monitoração
3. Cacti plugin - integrado com o cacti

“Neste artigo abordaremos a instalação do php-weathermap com a integração com o cacti.”

5.4 Iniciando a instalação

Entre no endereço abaixo:

<http://wotsit.thingy.com/haj/cacti/php-weathermap/download.html>

e baixe a versão atual: php-weathermap-0.82.zip

copie para dentro do diretório dos plugins do cacti.

Descompacte o arquivo

Unzip php-weathermap-0.82.zip

Depois digite o seguinte comando

```
mv php-weathermap-0.82.zip weathermap
```

5.5 Instalando como plugin do cacti

Antes de fazer qualquer outra coisa, verificar por favor que sua arquitetura Plugin está trabalhando corretamente com um plugin mais simples, como as ligações ou as ferramentas. Weathermap é relativamente complexo, e a busca de falhas sua arquitetura Plugin e Weathermap dos cactos ao mesmo tempo fará a vida mais dura para você!

Para usar como plugin do cacti, você *deve* descompactar os arquivos em um diretório chamado “/var/www/htdocs/cacti/plugins/weathermap”.

Você necessitará mudar as permissões no diretório da saída, de modo que o processo do poller dos cactos possa lhe escrever. Este é o mesmo como você faria para o diretório ao instalar cactos próprio do rra originalmente. Para um sistema do *nix, será algo como:

```
# chown cactiuser output
```

Então, editar seu arquivos do cacti em /var/www/htdocs/cacti/include/config.php e adicionar uma linha:

```
$plugins = array();  
$plugins[] = 'monitor';  
$plugins[] = 'weathermap';
```

Agora, atualize sua página do cacti, para ser certo que tudo está trabalhando correto. Se não, remova a linha que você adicionou apenas e você deve retornar ao normal.

Supondo a todos os olhares muito bem (mas não muito diferente), você podem começar permitir Weathermap. O início de uma sessão como o “admin” ou um outro usuário com direitas da gerência do usuário, vai à seção da gerência do usuário sob ferramentas no console dos cactos, e escolhe então seu próprio username da lista. Verificar as duas caixas novas dos “reinos” que deve haver - ver Weathermaps, e controlar Weathermaps - e estala então excepto. Uma aba de “Weathermap” deve aparecer no alto da página.

Antes que você comece a usar, você pôde querer mudar um ajuste de PHP. Weathermap usa um bocado justo da memória por padrões de PHP, porque constrói a imagem para o mapa na memória antes do saving ele. Em consequência, seu processo de PHP *pode* funcionar fora da memória. PHP tem de “um built-in da válvula segurança”, para parar certificados do runaway de matar seu usuário, que opta 8MB. Isto é controlado pelo “memory_limit =” linha em php.ini. Você pode necessitar aumentar este a 16MB ou a 32MB se você tiver problemas.

Aquele é ele! O Weathermap plugin é instalado. Para ir mais mais, você necessita algumas limas da configuração do weathermap definir seus mapas. Você pode fazer este em duas maneiras - usar o editor Correia-baseado do mapa, ou editando o texto baseou limas da configuração diretamente.

Para usar o editor, você necessita fazer algumas mais mudanças.

5.6 Configurando o editor

Se você estiver pretendendo usar o editor browser-baseado, então o diretório que você desembala a lima de fecho de correr no deve estar dentro da “do espaço correia fotorreceptora” na web server que funciona sua aplicação de levantamento de dados (isto é, cactos, MRTG, ou similar).

Começar seguindo as instruções acima para uma instalação Plugin autônoma ou dos cactos. Você será weathermap running apenas a mesma maneira, mas uma vez que você tem um weathermap trabalhando (ou plugin), você necessita continuar abaixo:

Copie a lima de editor-config.php-dist a editor-config.php. Se você usar cactos, e os quiser escolher origens de dados de sua instalação dos cactos pelo nome, editar a lima e certificar-se de que a linha que ajusta \$cacti_base está correta, e que o URI da base abaixo daquele está também correto para sua instalação dos cactos (estas duas linhas são MUDANÇA marcada na lima).

Certificar-se de que seu web server pode escrever ao diretório dos configs. Para fazer este, você necessita saber que usuário seu web server funciona como (talvez “ninguém”, “WWW” ou “httpd” em a maioria de *nixes) e funciona então:

configs do chown u+w dos configs de WWW do chown

Em um pinch, você pode apenas configs do chmod 777, mas esta não é realmente uma solução recomendada para um sistema de produção.

Em Windows, o mesmo aplica-se - o usuário que funciona o web server funciona como deve ter permissões escrever limas novas, e muda limas existentes no dobrador dos configs.

Você deve agora vai a <http://yourserver/wherever-you-unpacked-weathermap/editor.php> (ou a <http://yourserver/cacti/plugins/weathermap/editor.php>) em um browser, e começar uma página bem-vinda que ofereça à carga ou criar uma lima dos config. Aquele é ele. Feito toda. Ver por favor a [página manual do editor](#) para mais sobre *usar o editor*!

Nota importante da segurança: O editor permite *qualquer um* que pode alcançar editor.php para mudar as limas da configuração para seus weathermaps da rede. Não há nenhum built-in do authentication para editar, mesmo com os cactos Plugin. Isto é porque a lima da configuração não existe pelo defeito - o editor não trabalhará até que você escolha lhe fazer o trabalho. Recomendou que você qualquer um:

- mudar a posse de limas da configuração de modo que o editor não possa lhes escrever uma vez que estão completos, ou
- usar facilidades do controle do authentication e de acesso do seu web server ao limite que pode alcançar o URL de editor.php.

Antecipou que uma versão futura de Weathermap incluirá a integração melhor com authentication dos cactos para usuários da versão Plugin dos cactos deste editor.

5.7 Princípios Básicos

O weathermap é definido por uma lima do plain-text que pelo defeito seja chamada weathermap.conf (você pode ter muitas configurações, e escolhe entre elas com comando-linha interruptores). Pelo defeito, o weathermap lerá essa lima, e produzirá uma única lima do png chamada weathermap.png.

A lima da configuração tem três seções: Definições do nó, definições da ligação e ajustes globais. Há um [exemplo de uma lima completa no](#) fundo desta página.

Nodes

```
NODE nycore1
  LABEL NYC
  POSITION 30 30
```

A simple NODE

Os nós são os pontos em sua rede que são juntados junto. Dependendo do detalhe no mapa, puderam ser cidades ou routers individuais. Em um mapa básico, um nó tem 3 partes de informação - um nome interno que deva ser original a este nó, ele é posição do canto alto-esquerdo do mapa, nos pixels, e opcionalmente uma etiqueta, que apareça dentro da caixa que marca a posição do nó. Os nós sem uma etiqueta não aparecem no mapa em tudo, mas podem ainda ser usados como um endpoint para uma ligação.

Links

```
LINK backbone1
  NODES nycore1 paix1
  BANDWIDTH 3M
  TARGET ../my-mrtg-data/backbone1.html
```

A simple LINK

As ligações são as rotas da rede entre os nós. Tipicamente são ligações reais da rede, mas podem ser qualquer coisa que você pode começar a números para o esse faz o sentido no mapa.

Uma ligação mínima absoluta tem 3 partes de informação demasiado. São o nome interno original para esta ligação, e nomes originais do nó para os dois endpoints. Para mostrar o uso atual no mapa, você necessitará dar duas mais partes: a largura de faixa máxima na ligação, e uma maneira começar o throughput atual. A LARGURA DE FAIXA é medida nos bocados/segundo, e pode incluir os sufixos usuais de K, de M, de G e de T para valores grandes. A origem de dados é dada na linha de ALVO, e pode ser uma de

- lima MRTG-gerada do HTML (que contem um comentário especial do HTML no fundo com os valores atuais)
- lima Cacto-gerada de RRD
- Alguma outra lima de RRD, desde que você sabe está estruturada internamente.

Uma nota importante: a ordem dos nós na linha dos NÓS é significativa. O primeiro nó é considerado ser o "local" um ao pensar sobre a origem dos dados no ALVO. "para fora" estar "para fora" relativo ao primeiro nó. Se você encontrar o mapa mostra todos seus dados que fluem no sentido errado, tenta trocar a ordem dos nós aqui.

5.8 Ajustes globais

Estes ajustes vivem geralmente no alto da lima de texto, e especificam a informação básica sobre o mapa. Os ajustes mínimos são:

```
WIDTH 800
HEIGHT 600
```

Isto é especifica o tamanho do mapa nos pixels. Se você quiser algo um bocado mais extravagante do que um fundo branco liso, você pode compor uma imagem de fundo para usar-se no formato do png. Nesse caso, o mapa será o tamanho da imagem de fundo:

BACKGROUND western-europe.png

Há também os ajustes globais para ajustar-se que pias batismais são usadas para vários elementos do mapa, onde posicionar a cor-legenda, se em tudo, e que limas a output. A lista cheia está na seção da [referência](#) abaixo.

5.9 Uma Configuração de exemplo

Está aqui uma configuração da amostra, e [é aqui o que produz \(modificado para trabalhar um pouco melhor\)](#). Os dados não estão vivos na saída, mas dão-lhe uma idéia de o que pode ser feito. A imagem de fundo inicial está [aqui](#). A lima da configuração está também no diretório dos docs da distribuição, se você a quiser.

```
# some initial comments...
#
# This sample configuration file demonstrates most of the basic features of
# PHP Weathermap, along with some of the cosmetic and layout changes
# possible
#
#
BACKGROUND background.png
HTMLOUTPUTFILE example.html
IMAGEOUTPUTFILE example.png
TITLE Network Overview
HTMLSTYLE overlib
KEYPOS 10 400

# define some new TrueType fonts - built-in ones go from 1 to 5, so start high
FONTDEFINE 100 Veralt 8
FONTDEFINE 101 Vera 12
FONTDEFINE 102 Vera 9

KEYFONT 102

LINK DEFAULT
    BANDWIDTH 100M
    BWLABEL bits
    BWFONT 100
    OVERLIBWIDTH 395
    OVERLIBHEIGHT 153
    WIDTH 4

NODE DEFAULT
    LABELFONT 101

NODE transit
    POSITION 400 180
    LABEL TRANSIT

# a little splash of background colour for these nodes
NODE isp1
    POSITION 250 100
    LABEL ISP1
    INFOURL http://www.isp1.com/support/lookingglass.html
    LABELBGCOLOR 255 224 224
```

```
NODE isp2
    POSITION 550 100
    LABEL ISP2
    INFOURL http://www.isp2.net/portal/
    LABELBGCOLOR 224 255 224

NODE core
    POSITION 400 300
    LABEL core
    INFOURL https://core.mynet.net/admin/

NODE customer1
    LABEL xy.com
    POSITION 150 370

NODE customer2
    LABEL ww.co.uk
    POSITION 250 450

NODE infra
    LABEL INFRASTRUCTURE
    POSITION 450 450

# this node has an icon, and so we push the label to the South edge of it, so it
# can still be read
NODE sync
    LABEL Sync
    ICON my_router.png
    LABELOFFSET S
    LABELFONT 2
    POSITION 550 370
# the icon is taken from a Nagios icon pack:
#
http://www.nagiosexchange.org/Image\_Packs.75.0.html?&tx\_netnagext\_pi1\[p\_view\]=110&tx\_netnagext\_pi1\[page\]=10%3A10

NODE site1
    LABEL site1
    POSITION 700 220

NODE site2
    LABEL site2
    POSITION 750 420

LINK sync-core
    NODES sync core
    TARGET data/sync_traffic_in_259.rrd
    OVERLIBGRAPH
http://support.mynet.net/cacti/graph\_image.php?local\_graph\_id=256&rra\_id=0&graph\_nolegend=true&graph\_height=100&graph\_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=256
#
# Site1 has two E1s, so we use NODE-offsets to allow them to run parallel
#

LINK sync-site1a
    NODES sync:N site1:W
    WIDTH 3
    TARGET data/sync_traffic_in_257.rrd
    BANDWIDTH 2M
    OVERLIBGRAPH
http://support.mynet.net/cacti/graph\_image.php?local\_graph\_id=254&rra\_id=0&graph\_nolegend=true&graph\_height=100&graph\_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=126

LINK sync-site1b
    NODES sync:E site1:SE
    WIDTH 3
    OVERLIBGRAPH
http://support.mynet.net/cacti/graph\_image.php?local\_graph\_id=255&rra\_id=0&graph\_nolegend=true&graph\_height=100&graph\_width=300
    TARGET data/sync_traffic_in_258.rrd
    BANDWIDTH 2M
    INFOURL http://support.mynet.net/cacti/graph.php?rra\_id=all&local\_graph\_id=56
```

```
#
# site2 also has two links, but this time we use the VIA to curve the links
#
LINK sync-site2a
    NODES sync site2
    WIDTH 3
    VIA 650 380
    TARGET data/sync_traffic_in_251.rrd
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=248&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    BANDWIDTH 1M
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=252

LINK sync-site2b
    NODES sync site2
    WIDTH 3
    VIA 650 420
    TARGET data/sync_traffic_in_252.rrd
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=228&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    BANDWIDTH 1M
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=561

#
# ISP 1 has a several links, again, but they prefer to see one arrow, and the aggregate bandwidth
# so we use multiple TARGETs on one line, here, to sum the data

LINK transit-isp1
    NODES transit isp1
    TARGET data/trans1_traffic_in_352.rrd data/trans1_traffic_in_378.rrd data/trans1_traffic_in_420.rrd
    BANDWIDTH 10M
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=355&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=633

LINK transit-isp2
    NODES transit isp2
    TARGET data/trans1_traffic_in_438.rrd
    BANDWIDTH 34M
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=433&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=265

LINK core-transit
    NODES transit core
    TARGET data/trans1_traffic_in_350.rrd
    ARROWSTYLE compact
    WIDTH 4
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=347&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=122

LINK cust1-core
    NODES customer1 core
    TARGET data/extreme_traffic_in_299.rrd
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=296&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=237

LINK cust2-core
    NODES customer2 core
    TARGET data/extreme_traffic_in_286.rrd
    OVERLIBGRAPH
    http://support.mynet.net/cacti/graph_image.php?local_graph_id=283&rra_id=0&graph_nolegend=true&graph_height=10
    0&graph_width=300
    INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=222

LINK infra-core
    NODES infra core
    TARGET data/extreme_traffic_in_294.rrd
```

OVERLIBGRAPH

http://support.mynet.net/cacti/graph_image.php?local_graph_id=291&rra_id=0&graph_nolegend=true&graph_height=100&graph_width=300

INFOURL http://support.mynet.net/cacti/graph.php?rra_id=all&local_graph_id=228

6. Instalando e Configurando o Smokeping

6.1 Introdução

SmokePing é uma ferramenta deluxe da medida da latência. Pode medir, armazenar e indicar a latência, a distribuição da latência e a perda do pacote. SmokePing usa RRDtool manter um a longo prazo dados-armazena e extrair os gráficos bonitos, dando até a informação minuciosa no estado de cada conexão de rede, SmokePing usa encaixes da medida da latência para o extendability sem emenda.

6.2 Alarmes espertos

SmokePing vem com um sistema de alarme esperto. Aparte dos alarmes simples do ponto inicial, você tem a opção de definir testes padrões da latência ou da perda e usa-os provocar alarmes. Isto permite que você defina um teste padrão que gere um único alarme quando a perda vai de abaixo 1% sobre a 20% e permanece sobre 20% por mais de 10 minutos. A vantagem desta aproximação é o elimination virtual dos alarmes duplicados que você começaria com ponto inicial simples um sistema baseado.

Os alarmes podem ser emitidos a um endereço do correio ou um pager e se você quiser você pode também começar um certificado externo segurar os alarmes.

6.3 Sustentação dinâmica do IP

SmokePing pode também tratar das máquinas que mudam freqüentemente seu IP address permitindo que o anfitrião remoto chame a atenção de SmokePings a seu IP address novo. Em tal instalação SmokePing monitora também para quanto tempo o sistema remoto poderia manter seu IP address. Para assegurar-se de que não esteja monitorando de repente o anfitrião errado, SmokePing tenta mais mais fingerprint cada um de seus alvos dinâmicos do IP através do SNMP.

6.4 Fundo técnico

SmokePing é escrito no Perl. Consiste em um processo do daemon responsável para o levantamento de dados e em um certificado do cgi que apresenta os dados na correia fotorreceptora. O certificado do cgi usa SpeedyCGI conseguir o mod_perl como o desempenho sem a necessidade desdobrar o mod_perl em seu web server. SmokePing ostenta uma arquitetura de encaixe para adicionar facilmente potencialidades novas da medida da latência a SmokePing. SmokePing trabalha feliz em todas as plataformas do Unix.

6.5 Pré-requisitos

SmokePing não é autônomo. Confia em várias outras ferramentas e serviços que estão atuais. Além de um sistema do Unix e de uma instalação trabalhando do Perl você necessita as seguintes coisas. A lista contém os nomes das ferramentas junto com a versão da ferramenta que eu estou usando, e uma ligação para downloading a ferramenta.

RRDtool 1.0.x ou 1.2.x

<http://oss.oetiker.ch/rrdtool/>

Seguir apenas as instruções de instalação. Nós necessitamos este todo do pacote todo o registrar e representar graficamente em SmokePing.

FPing

<http://www.fping.com/>

Você deve agarrar a versão a mais nova, 2.4b2_to-ipv6, porque tem a sustentação IPv6 e realmente as configurações, ao contrário das mais velhas.

EchoPing (opcional)

<http://echoping.sourceforge.net/>

Você necessita este funcionar as pontas de prova de EchoPing

Curl (Optional)

<http://curl.haxx.se/>

Você necessita este para a ponta de prova da onda.

Dig (Optional)

<http://www.isc.org/sw/bind/>

Você necessita este para a ponta de prova do DNS.

SSH (opcional)

<http://www.openssh.org/>

Você necessita este para a ponta de prova de SSH.

Vários módulos do Perl (opcionais)

Você pode começar estes de <http://www.cpan.org/>.

Socket6 0.11-1

Opcionalmente instalar o módulo Socket6 para o Perl permite smokeping para verificar os hostnames que resolvem somente a um endereço IPv6. Se você sondar tais anfitriões que usam o IE. a ponta de prova FPing6 e começa os avisos que aqueles anfitriões não resolvem a um IP address, você necessita instalá-lo.

Rede:: Telnet

Você necessita este para a ponta de prova de TelnetIOSPing.

Rede:: DNS

Você necessita este para a ponta de prova de AnotherDNS.

Rede:: LDAP

Você necessita este para a ponta de prova de LDAP.

IO:: Socket:: SSL

Você necessita este se você quiser a ponta de prova de LDAP usar o comando dos "starttls".

Authen:: Radius

Você necessita este para a ponta de prova do raio.

Web server

<http://httpd.apache.org/>

O poço I não começará muito neste. A coisa importante é, para ter um web server que permita que você funcione certificados do cgi. Se você estiver usando Apache mim fortemente recomendar usar o sistema do suexec para funcionar certificados do cgi. Frequentemente é suficiente mudar o suexec binário à raiz do setuid e reiniciar Apache. Usando o suexec reservar para funcionar certificados do cgi sob clientes apropriados do usuário.

Perl 5.6.1

<http://www.perl.com>

Eu suponho que você começará afastado com versões mais velhas do Perl. Eu estou usando 5.6.1 aqui e trabalha muito bem.

SpeedyCGI

<http://www.daemoninc.com/SpeedyCGI/>

SpeedyCGI apressa-se acima de CGIs escrito no Perl dramática fazendo lhes o residente da memória e entregando o pedido novo ao certificado que está funcionando já. SmokePing optimized para o uso com SpeedyCGI. Anotar que você não necessitam mod_speedy, apenas o liso e executável speedy simples, este é a beleza desta ferramenta que trabalha sem tocar em seu apache... de outra maneira você poderia também usar FastCGI ou mod_perl. Há um erro em speedy com mensagens de erro de retorno ao registro de erro de apache. Você pode encontrar um remendo na área smokeping do download.

Cgi:: Carp

<http://http://search.cpan.org/~lds/CGI.pm/>

Se você estiver usando uma versão de Perl mais velha de 5.8.1, você deve considerar instalar uma versão mais nova do cgi:: Carpa (devem ser pelo menos 1.24, incluído em CGI.pm-2.82). Se não nenhuma mensagens de erro geradas pelo certificado do cgi não mostrarão acima completamente tão limpo como poderiam no registro de erro do web server. Se você não puder nem não quiser promover o cgi:: Carpa, você pode também ajustar o changecgiprogramname variável na seção geral ao no. do valor.

6.6 Instalação

Uma vez que as ferramentas alistadas acima estão no lugar, você pode começar ajustar-se - acima de SmokePing próprio. Uma vez que as ferramentas de SmokePing são desembaladas, são mais de mais menos pronto para usar-se. Pelo menos à extensão, isso não é necessário instalá-los. Você pode querer rebatizar o diretório a um nome que não contenha o número de versão de SmokePing, ou faz pelo menos um symlink, de modo que você possa usar um nome persistente ao consultar às limas de SmokePing.

6.7 Requerimentos

- HTTP- Servidor
- SmokePing- v2.0.4
- RRDtool- v1.2.11
- cgilib- v0.5
- fping- v2.4b2

6.8 Instalação do fping

<http://web162.servana.de/arquivos/fping.tar.gz>

```
# tar -xzf fping.tar.gz
```

```
# cd fping -2.4b2_to
```

```
# ./configure
```

```
# make
```

```
# make install
```

6.9 Instalação do cgi

<http://web162.servana.de/ferramentas/cgilib-0.5.tar.gz>

-

```
cgilib-0.5 entpacken
```

```
# tar -xzf cgilib -0.5.tar.gz
```

```
# cd cgilib -0.5
```

```
# make
```

6.10 Instalando o RRDTOOL

Entra na pagina:

<http://linuxpackages.telecoms.bg/Slackware-10.1/Console/rrdtool/rrdtool-1.2.9-i486-2kf.tgz>

```
installpkg rrdtool-1.2.9-i486-2kf.tgz
```

6.11 Instalação Smokeping

```
# http://web162.servana.de/arquivos/smokeping-2.0.4.tar.gz
```

```
# tar -xzf smokeping -2.0.4.tar.gz
```

```
# mv smokeping -2.0.4 /usr/local/smokeping
```

```
# cd /usr/local/smokeping/bin
```

```
# mv smokeping.dist smokeping
```

6.12 Configurando

```
vi /bin/smokeping
```

Altere as configurações destacadas de acordo com as suas configurações locais:

```
#!/usr/bin/perl -w
# *-perl*-
```

```
use lib qw(/usr/bin/rrdtool/lib/perl);
use lib qw(/usr/local/smokeping/lib);
```

```
use Smokeping 2.000004;
```

```
Smokepingmain::("/usr/local/smokeping/etc/config");
```

```
# cd /usr/local/smokeping/htdocs
# mv smokeping.cgi.dist smokeping.cgi
```

```
# vi smokeping.cgi
```

```
#!/usr/bin/perl -w
# *-perl*-
```

```
use lib qw(/usr/local/rrdtool/lib/perl);
```

```
uso lib qw(/usr/local/smokeping/lib);
uso CGI::Carp:: qw(fatalsToBrowser());

uso Smokeping 2.000004;

Smokeping::Cgi::("/usr/local/smokeping/etc/config");
```

6.13 SmokePing Configuração

Código:

```
# cd /usr/local/smokeping/etc
# mv config.dist config
# mv tmail.dist tmail
# mv smokemail.dist smokemail
# mv basepage.html.dist base.html

# vi config

***geral ***

# Hauptkonfiguration
titular= MyName
contato= MyName@MyCompany.com
mailhost = smtp.company.com
sendmail = /usr/lib/sendmail
imgcache = /var/www/htdocs/.simg
imgurl = ../.simg
datadir = /usr/local/smokeping/var
piddir = /usr/local/smokeping/var
cgiurl = http://localhost/cgi-bin/smokeping.cgi
smokemail = /usr/local/smokeping/etc/smokemail
tmail = /usr/local/smokeping/etc/tmail
syslogfacility = local0
# concurrentprobes = no

*** Alerts ***
to = puppet@gmx.org
from = puppet@gmx.org

+bigloss
type = loss
# in percent
pattern = ==0%,==0%,==0%,==0%,>0%,>0%,>0%
comment = suddenly there is packet loss

+someloss
type = loss
```

```
# in percent
pattern = >0%,*12*,>0%,*12*,>0%
comment = loss 3 times in a row

+startloss
type = loss
# in percent
pattern = ==S,>0%,>0%,>0%
comment = loss at startup

+rttdetect
type = rtt
# in milli seconds
pattern = <10,<10,<10,<10,<10,<100,>100,>100,>100
comment = routing messed up again ?

*** Database ***
step = 120
pings = 10

# consfn mrhb steps total

AVERAGE 0.5 1 1008
AVERAGE 0.5 12 4320
MIN 0.5 12 4320
MAX 0.5 12 4320
AVERAGE 0.5 144 720
MAX 0.5 144 720
MIN 0.5 144 720

*** Presentation ***

template = /usr/local/smokeping/etc/basepage.html

+ overview

width = 600
height = 50
range = 12h

+ detail

width = 600
height = 200
unison_tolerance = 2

"Last 3 Hours" 3h
"Last 24 Hours" 24h
"Last 7 Days" 7d
"Last 30 Days" 30d
```

"Last 365 Days" 365d

*** Probes ***

+ FPing

binary = /usr/local/sbin/fping

*** Targets ***

probe = FPing

menu = Top

title = Latenz im LAN

remark = Bitte eine Gruppe auf der linken Seite auswaehlen

+ LAN

menu = Lokal

title = Lokales Netzwerk

++ Fileserver

menu = Fileserver

title = Datenlager

host = 192.168.0.1

++ Gatekeeper

menu = Gatekeeper

title = Der Tuersteher

host = gatekeeper.tovdc.local

+ WAN

menu = Internet

title = Internet

++ NetworkLab

menu = NetworkLab

title = NetworkLab - Das Netzwerkportal

host = www.nwlab.net

Salve, vamos para o próximo passo, criar os diretórios necessários:

```
# mkdir /var/www/htdocs/.simg
```

```
# chmod777 /var/www/htdocs/.simg
```

```
# mkdir /usr/local/smokeping/var
```

```
# mkdir /var/www/htdocs/cgi-bin
```

```
# cp /usr/local/smokeping/htdocs/smokeping.cgi /var/www/htdocs/cgi-bin
```

```
# chmod 755 /var/www/htdocs/cgi-bin/smokeping.cgi
```

Configurado o Apache /etc/apache/httpd.conf

Executando:

```
# /usr/local/smokeping/bin/smokeping
```

```
http://localhost/cgi-bin/smokeping.cgi/
```

```
# pico /etc/rc.d/rc.local
```

Código:

```
echoStartingSmokePing " : /usr/ local/smokeping/bin/smokeping"  
usr/local/smokeping/bin/mokeping
```

```
Código: /usr/local/smokeping/bin/smokeping -reload
```


7. Considerações Finais

Através desse controle é possível determinar e saber em tempo real tudo que está acontecendo na rede da empresa, determinando as ações e providências que devem ser tomadas evitando problemas maiores.

Existem inúmeras vantagens ao adotar ferramentas de monitoramento destacando o controle que fornecem ao administrador. Em primeiro lugar é possível saber o que está acontecendo na rede e ainda em tempo real, assim é possível identificar para onde a ação será direcionada.

Um dos principais objetivos em monitorar uma rede é permitir ações preventivas às possíveis paralisações, garantindo uma estrutura estável para a empresa, ações pró-ativas que podem diminuir as influências externas gerando maior produtividade é o que as empresas e organizações buscam para obterem resultados positivos em seus focos de negócio. mais informações mande um e-mail para: emerson.souza@pmm.am.gov.br

8. Referencias Bibliográficas

<http://wotsit.thingy.com/haj/cacti/php-weathermap/>

<http://www.nagios.org/>

<http://www.cacti.net/>

<http://oss.oetiker.ch/smokeping/>

OUTRAS APOSTILAS EM:
www.projetoderedes.com.br