Grupo de Resposta a Incidentes de Segurança – DCC/UFRJ

Última Modificação: 29/06/09

MONIT - Aplicativo para Administração Segura com Alertas

GRIS-2009-T-001

Universidade Federal do Rio de Janeiro
Instituto de Matemática
Departamento de Ciência da Computação
Grupo de Resposta a Incidentes de Segurança

Rio de Janeiro, RJ – Brasil

MONIT – Aplicativo para Administração Segura com Alertas

GRIS-2009-T-001

Guilherme Alves Cardoso Penha

A versão mais recente deste documento pode ser obtida na página oficial do GRIS

GRIS – Grupo de Resposta a Incidentes de Segurança
UFRJ - Universidade Federal do Rio de Janeiro
Decania do CCMN - Centro de Ciências Matemáticas e da Natureza
Av. Athos da Silveira Ramos, s/n
Cidade Universitária - Rio de Janeiro/RJ
CEP: 21941-590

Sumário

1. Introdução

O Monit é uma ferramenta que auxilia administradores de servidores na verificação periódica da situação de seus sistemas, analisando elementos como execução de programas, consumo de memória ou CPU, permissão de arquivos e até mesmo a modificação de arquivos, pastas ou dispositivos. Através de sua administração via Web, é possível iniciar, parar e reiniciar processos, assim como ativar e desativar o monitoramento.

O Monit é uma ferramenta de fácil instalação, configuração, utilização e está disponível diretamente no sistema de pacotes das principais distribuições Linux e BSD, tais como Debian/Ubuntu, RHEL/Fedora/CentOS, SuSE, Slackware, Gentoo, Mandriva, Free/Net/OpenBSD, entre outros.

Existem diversas ferramentas hoje no mercado que realizam tarefas similares à do Monit. Porém, além do fato de ser uma ferramenta OpenSource, o Monit possui uma grande flexibilidade de configuração, facilidade de implementação, manutenção e grande eficiência.

2. Instalando o Monit

Há duas formas essenciais de se instalar o Monit: a partir do código fonte ou através de pacotes pré-compilados. Inicialmente, vamos explicar como é o processo de instalação a partir do código fonte. Em seguida, indicaremos os comandos necessários para a instalação de pacotes em diferentes distribuições Linux e BSD. A utilização de gerenciadores de pacotes, quando disponíveis em seu sistema, é recomendada por questões de praticidade e administração. Deve-se levar em conta, no entanto, quão confiável é o repositório de pacotes e quão atualizada está a versão do software.

2.1. Instalação a partir do código-fonte

Obtenha a versão estável mais atual na página oficial da ferramenta (*http://mmonit.com/monit/*). No exemplo abaixo, utilizamos a ferramenta wget para obtenção do arquivo monit-x.xx.x.tar.gz, onde x.xx.x é a versão desejada:

```
$ wget http://mmonit.com/monit/dist/monit-x.xx.x.tar.gz
```

Descompacte o código com o comando “tar”. A pasta “monit-x.xx.x” será criada com o conteúdo expandido:

```
$ tar zxvf monit-x.xx.x.tar.gz
```

Entre na pasta criada:

```
$ cd monit-x.xx.x
```

Para compilar o Monit, execute o comando abaixo:

```
$ ./configure
```

Esse comando cria um arquivo de compilação (Makefile) de acordo com o seu sistema. Caso encontre problemas durante a execução deste comando ou deseje personalizar as opções de compilação e instalação (como caminho para bibliotecas e diretório de destino), veja as opções disponíveis através do comando *“./configure –-help”*. Em uma instalação padrão, o Monit é instalado no diretório /usr/local/bin/. Você pode alterar este destino através do parâmetro –prefix no momento do ./configure.

Para realizar a compilação, use o comando make:

```
$ make
```

Após sucesso na compilação, basta instalar no sistema com o comando abaixo (executado como superusuário, ou “*root*”):

```
# make install
```

Agora que o Monit já está instalado, podemos iniciar a parte de configuração e personalização da ferramenta.

2.2. Pacotes pré-compilados

Como mencionado anteriormente, outra forma de instalarmos o Monit é através de pacotes pré-compilados. Para maiores informações sobre a instalação de programas, consulte a documentação da sua distribuição.

Para a instalação em sistemas Debian-like (tal como Ubuntu), podemos usar o apt-get para o modo texto:

```
# apt-get install monit
```

Para a instalação em sistemas RHEL-like (tais como CentOS/Fedora) e SuSE, podemos usar o Yum para o modo texto:

```
# yum install monit
```

Para a instalação em sistemas BSD-like (como FreeBSD/OpenBSD), podemos usar o PKG, assim como o Ports. Para os comando abaixo, assume-se que a URL com os aplicativos disponíveis do PKG esteja correta e que o caminho da árvore do Ports seja /usr/ports.

1ª alternativa (PKG):

```
# pkg_add -v monit-x.xx.xx.tgz
```

2ª alternativa (Ports):

```
# cd /usr/ports/net/monit
# make install
# pkg_add -v monit-x.xx.xx.tgz
```

3. Configurando o Monit

Após sucesso na instalação, iremos editar o arquivo de configuração do Monit, chamado "monitrc". Mas antes disto, faremos uma cópia de segurança do arquivo original, que fica normalmente em /etc/monit, com o comando:

```
# cp /etc/monit/monitrc  /etc/monit/monitrc.original
```

O primeiro passo é ajustarmos a frequência com que o Monit será executado, a porta que ele escutará, assim como usuário e senha para o acesso:

```
set daemon 120
set httpd port 2812 and
      allow USUARIO_ADMIN:SENHA_ADMIN
      allow OUTRO_USUARIO:OUTRA_SENHA read-only
```

O parâmetro *daemon* controla a freqüência de execução do Monit, enquanto o *httpd port* controla a porta em que será executado um servidor web minimalista destinado à administração remota da ferramenta. O parâmetro *allow* indica o par usuário/senha requerido para o acesso através da Web[1]. O parâmetro *read-only* após o par usuário/senha desabilita o acesso aos botões de iniciar, parar e reiniciar os processos e monitoramentos.

A forma padrão de conexão à interface administrativa do Monit é por HTTP, porém é fortemente recomendado habilitar o SSL, de forma que a conexão entre o Monit e o navegador Web seja criptografada. Caso não tenha um certificado SSL para o seu site, consulte a seção de dicas do site do GRIS e crie o seu certificado.

1 Note que a senha será armazenada no arquivo de configuração em texto puro, então qualquer pessoa com permissão de leitura para este arquivo saberá a senha de administração do Monit. É altamente recomendado que as permissões do arquivo de configuração do Monit sejam ajustadas para permitir leitura apenas pelo administrador, ou por pessoas de um grupo específico com privilégios, como "monit", por exemplo.

É altamente recomendada a utilização de uma senha forte. O GRIS disponibiliza em sua seção de artigos um documento ensinando como escolher senhas. O link é www.gris.dcc.ufrj.br/artigos.php.

Aqui segue um exemplo de implementação do SSL no Monit para uma conexão criptografada, supondo que o certificado seja /etc/monit/monit.pem.

```
SSL ENABLE
      PEMFILE /etc/monit/monit.pem
```

Devemos configurar também: em *logfile*, o tipo de registro(log) que será gerado pelo Monit; em *alert*, o email para onde alertas serão enviados; e em *mailserver*, o servidor de email que será utilizado para enviar as mensagens. Note que estas opções não são obrigatórias, mas são de grande importância para o monitoramento efetivo do sistema. O *mail-format* é apenas uma maneira de ajustar as características do email a ser enviado.

```
set logfile syslog facility log_daemon
set alert monitor@meu.servidor.de.email
set mailserver meu.servidor.de.email
set mail-format{ from:monit@maquina }
```

Após essas edições, devemos configurar os processos e arquivos que deverão ser monitorados pelo Monit.

Outros exemplos oficiais podem ser encontrados no seguinte endereço:

http://mmonit.com/monit/documentation/monit.html#configuration_examples

Agora segue uma breve descrição dos parâmetros utilizados para a configuração do monitoramento de aplicativos:

O exemplo será dado com a verificação de um processo com o nome NOME_PROCESSO e com o pid do processo localizado em CAMINHO_PID_PROCESSO.

```
check process NOME_PROCESSO with pidfile CAMINHO_PID_PROCESSO
```

Abaixo segue os comando para a inicialização e finalização do processo analisado:

```
start program = COMANDO_INICIALIZAÇÃO_PROGRAMA
stop program = COMANDO_FINALIZAÇÃO_PROGRAMA
```

Nesta parte trataremos das condições de consumo de recursos, tais como CPU e memória. Para isto assumiremos que o processo pode consumir até o VALOR_EM_PORCENTAGEM de CPU por no máximo um numero OCORRÊNCIAS de vezes. Caso alguma destas regras seja atingida, a ação alert ou restart será executada. É importante ressaltar que o valor de ocorrências para o reinicio deve ser maior que o valor para alerta. Devemos considerar também o valor do consumo de memória. Neste caso o Monit deve somente alertar caso o processo consuma mais do que o valor de CONSUMO_MEMÓRIA.

```
if cpu > VALOR_EM_PORCENTAGEM% for OCORRÊNCIAS cycles then alert
if cpu > VALOR_EM_PORCENTAGEM% for OCORRÊNCIAS cycles then restart
if totalmem > CONSUMO_MEMORIA then alert
```

Este teste é baseado em conexões para o serviço. Para isto ele testa uma conexão na porta NUMERO_PORTA com o protocolo PROTOCOLO_PROGRAMA referente ao aplicativo monitorado. Caso não obtenha sucesso, ele reinicia o serviço. Repare que esta regra pode ser adaptada a inúmeros protocolos e estes devem ser consultados no website oficial ou no manual do Monit por ser uma particularidade de cada versão do programa.

```
if failed port NUMERO_PORTA protocol PROTOCOLO_PROGRAMA then restart
```

Caso o Monit reinicie o processo por um numero TENTATIVAS de vezes com uma número OCORRENCIAS de ciclos, é indicado que ele pare de monitorar o processo.

```
if TENTATIVAS restarts within OCORRÊNCIAS cycles then timeout
```

Um detalhe que deve ser observado é o fato de que todos os processos estão exemplificados a seguir com seus caminhos e portas padrões. Sempre que possível, altere tais campos para evitar ataques automatizados.

3.1. OpenSSH


```
# seção para o ssh
check process ssh with pidfile /var/run/sshd.pid
      start program = "/etc/init.d/ssh start"
      stop program = "/etc/init.d/ssh stop"
      if cpu > 10% for 2 cycles then alert
      if cpu > 20% for 3 cycles then restart
      if totalmem > 20Mb then alert
      if failed port 22 protocol ssh then restart
      if 5 restarts within 5 cycles then timeout
```

3.2. Apache


```
# seção para o apache 2
check process apache with pidfile /var/run/apache2.pid
      start program = "/usr/sbin/apache2ctl start"
      stop program = "/usr/sbin/apache2ctl stop"
      if cpu usage > 60% for 2 cycles then alert
      if cpu usage > 80% for 5 cycles then restart
      if totalmem > 100Mb then alert
      if failed port 80 protocol http
            with timeout 15 seconds
            then restart
      if failed port 443 type tcpssl protocol http
            with timeout 30 seconds
            then restart
      if 5 restarts within 5 cycles then timeout
```

3.3. MySQL

```
# seção para o MySQL
check process mysql with pidfile /var/run/mysqld/mysqld.pid
      start program = "/etc/init.d/mysql start"
      stop program = "/etc/init.d/mysql stop"
      if cpu usage > 30% for 2 cycles then alert
      if cpu usage > 70% for 5 cycles then restart
      if totalmem > 100Mb then alert
      if failed port 3306 protocol mysql then restart
      if 5 restarts within 5 cycles then timeout
```

3.4. VSFTP

```
# seção para vsftp
check process vsftpd with pidfile /var/run/vsftpd/vsftpd.pid
      start program = "/etc/init.d/vsftpd start"
      stop program = "/etc/init.d/vsftpd stop"
      if failed port 21 protocol ftp then restart
      if totalmem > 50Mb then alert
      if cpu usage > 80% for 3 cycles then restart
      if 5 restarts within 5 cycles then timeout
```

3.5. Postfix


```
# seção para o postfix
check process postfix with pidfile /var/spool/postfix/pid/master.pid
      start program = "/etc/init.d/postfix start"
      stop program = "/etc/init.d/postfix stop"
      if failed port 25 protocol smtp then restart
      if totalmem > 80Mb then alert
      if 5 restarts within 5 cycles then timeout
```

4. Monitorando Arquivos

O Monit permite ainda o monitoramento de arquivos em seu sistema. No exemplo abaixo, monitoramos o arquivo de regras do firewall, localizado em /etc/init.d/firewall. O exemplo pode, naturalmente, ser expandido para monitorar quantos arquivos forem necessários.

De forma similar ao monitoramento de processos, no caso de arquivos informaremos o caminho do arquivo a ser monitorado na cláusula *check,* informando o nome do arquivo e o caminho do mesmo. Observe que o que importa é o caminho completo para o arquivo e o nome informado após o *file* é somente ilustrativo, podendo ser qualquer nome.

```
# seção para monitoramento do arquivo de regras do firewall
check file firewall with path /etc/init.d/firewall
```

Neste ponto é analisado o proprietário, o grupo pertencente, as permissões e o MD5 do arquivo. Caso alguma destas verificações falhe, ele simplesmente para de monitorar e em outro ponto desta seção ele realiza a ação desejada.

```
      if failed uid root then unmonitor
      if failed gid root then unmonitor
      if failed permission 711 then unmonitor
      if failed checksum
            and expect the sum  58e4f20902be3d04cd715b323771fa09
            then unmonitor
```

Após realizar os testes acima, ele realiza a ação alert caso algum teste dos itens tenha falhado. Neste setor é apresentado uma maneira alternativa de personalizar o email de alerta, onde aplica-se um assunto diferente a fim de ressaltar a diferença com o email padrão de alerta definido anteriormente.

```
alert monitor@meu.servidor.de.email on {
      permission, uid, gid, unmonitor
} with the mail-format { subject: Alarm! }
```

5. Monitorando Recursos de Sistemas

Nesta seção encontramos exemplos fictícios de um monitoramento em um sistema UNIX. É interessante ressaltar que estes valores devem ser adaptados de acordo com a sua necessidade de disponibilidade de recursos de sua máquina.

No exemplo para monitoramento local definimos o nome do sistema para uma melhor visualização na interface do Monit. Analisamos o load, a utilização de memória e o uso de CPU pelo sistema. Todos estes itens devem ser definidos para alertar em caso de problemas.

```
# seção para um sistema local
check system endereco.do.site
      if loadavg (1 min) > 4 then alert
      if loadavg (5 min) > 2 then alert
      if memory usage > 90% then alert
      if cpu usage (user) > 70% then alert
      if cpu usage (system) > 70% then alert
      if cpu usage (wait) > 30% then alert
```

Para o exemplo de monitoramento remoto, definimos a cláusula *check* como no monitoramento de arquivos, porém aqui alteramos o *file* por *address.*

```
# seção para um sistema remoto
check host nome.host.remoto with address endereço.do.host.remoto
```

Neste ponto testaremos a conexão de 3 maneiras. São elas o envio de ping's com o tempo de expiração de 15 segundos, o teste de acesso pela web com o protocolo http e a última maneira acessando pela web com o protocolo https. É importante ressaltar que para o sucesso destes testes, as requisições de ping para a máquina devem ser aceitas e que o arquivo definido na cláusula *request* deve ser um arquivo válido.

```
if failed icmp type echo count 5 with timeout 15 seconds
      then alert
if failed port 80 protocol http
      and request “index.html”
      then alert
if failed port 443 type tcpssl protocol http
      and request “index.html”
      then alert
```

6. Execução Básica

6.1. Verificando o arquivo de configuração

É muito importante testar o arquivo de configuração antes de carregá-lo para o sistema. Para isto, basta executar o comando:

```
# monit -t
```

A saída esperada é “*Control file syntax OK”*.

Caso apareça algum erro, o Monit indicará um mensagem contendo a linha onde o erro foi encontrado. Após isso, analise cuidadosamente o arquivo de configuração do programa e corrija o necessário. No caso abaixo é somente um erro na formatação do email.

```
[monit: parse error 'usuarioservidor.com.br' at line 85]
```

6.2. Executando o Monit

Para carregar o arquivo de configuração desejado, execute o seguinte comando informando o caminho do arquivo:

```
# monit -c /usr/local/etc/monitrc
```

Após a primeira inicialização informando o arquivo de configuração, pode-se iniciar o Monit com o comando abaixo:

```
# monit
```

6.3. Encerrando a Execução do Monit

Para finalizar o daemon do Monit, execute:

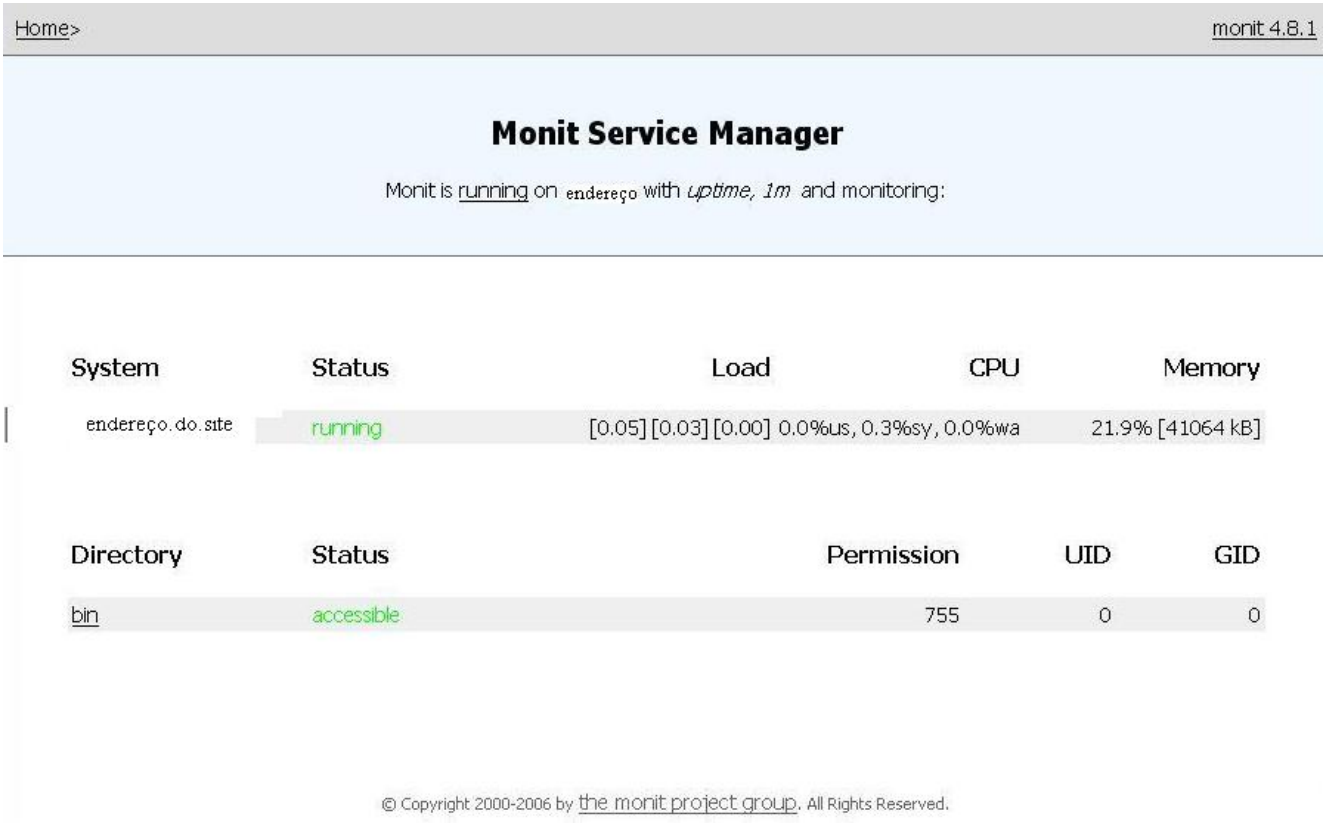
```
# monit quit
```

7. Administrando Remotamente

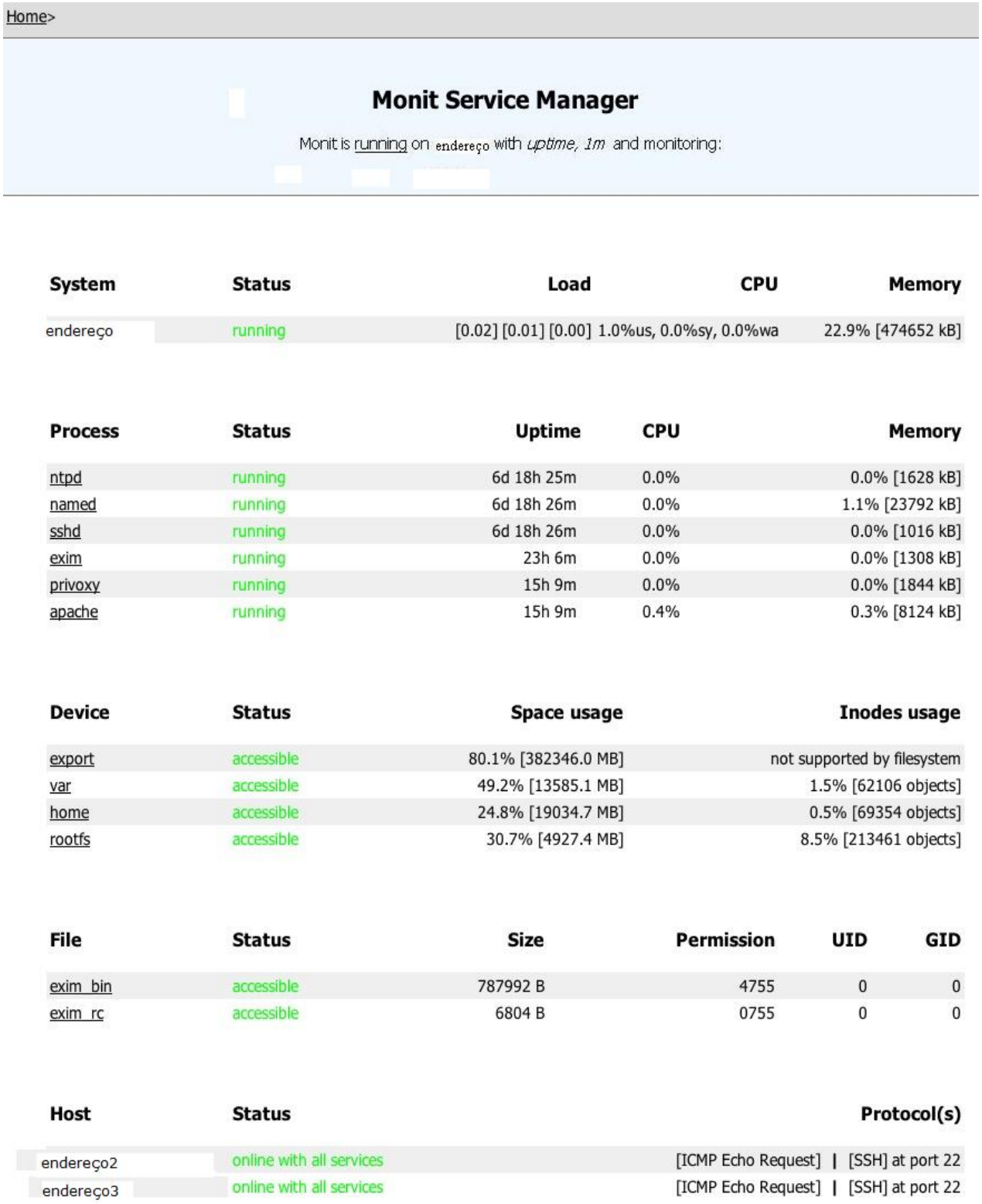
Para efetuar a administração remota, acesse o seu navegador, escolha entre http e https, insira o endereço da máquina onde o Monit está instalado e a porta definida no arquivo de configuração.

No caso deste tutorial, por exemplo, ficaríamos com *https://endereco.do.site:2812*

Exemplo de tela inicial:



Outro exemplo de tela inicial:



Home>

Monit Service Manager

Monit is running on endereço with *uptime, 1m* and monitoring:

System	Status	Load	CPU	Memory
endereço	running	[0.02] [0.01] [0.00]	1.0%us, 0.0%sy, 0.0%wa	22.9% [474652 kB]

Process	Status	Uptime	CPU	Memory
ntpd	running	6d 18h 25m	0.0%	0.0% [1628 kB]
named	running	6d 18h 26m	0.0%	1.1% [23792 kB]
sshd	running	6d 18h 26m	0.0%	0.0% [1016 kB]
exim	running	23h 6m	0.0%	0.0% [1308 kB]
privoxy	running	15h 9m	0.0%	0.0% [1844 kB]
apache	running	15h 9m	0.4%	0.3% [8124 kB]

Device	Status	Space usage	Inodes usage
export	accessible	80.1% [382346.0 MB]	not supported by filesystem
var	accessible	49.2% [13585.1 MB]	1.5% [62106 objects]
home	accessible	24.8% [19034.7 MB]	0.5% [69354 objects]
rootfs	accessible	30.7% [4927.4 MB]	8.5% [213461 objects]

File	Status	Size	Permission	UID	GID
exim_bin	accessible	787992 B	4755	0	0
exim_rc	accessible	6804 B	0755	0	0

Host	Status	Protocol(s)
endereço2	online with all services	[ICMP Echo Request] | [SSH] at port 22
endereço3	online with all services	[ICMP Echo Request] | [SSH] at port 22

8. Dúvidas e Soluções

Caso tenha dúvidas em assuntos não abordados neste documento, visite o FAQ oficial, assim como lista de discussão em http://mmonit.com/wiki/ .

Caso tenha algo a acrescentar neste documento, não hesite em contatar o GRIS. Estaremos sempre dispostos a melhorar nossos materiais e colaborar na medida do possível com a comunidade.

9. Referências Bibliográficas

Página oficial do projeto Monit

http://mmonit.com/monit/

Página com ajudas sobre o Debian disponibilizadas pela comunidade

http://www.debianhelp.co.uk/monit.htm

Lista de discussão oficial da comunidade do Monit

http://mail.freesoftware.fsf.org/mailman/listinfo/monit-general