

Integrar tomcat php4 munin

Esta documentação faz a configuração do NGINX com PHP4 usando CGI e fazendo redirect dos sistemas feitos em Maker para o Tomcat.

Autor: Gustavo Soares <slot.mg@gmail.com>

CONFIGURAÇÃO PARA NGINX e PHP4 usando bind.

Adicionar a chave e o repositório do NGINX.

```
root@ecasp:/root> wget -q0 - http://nginx.org/keys/nginx_signing.key | apt-key add -
root@ecasp:/root> echo "deb http://nginx.org/packages/debian/ $(lsb_release -sc) nginx" > /etc/apt/
sources.list.d/nginx.list
root@ecasp:/root> /etc/init.d/apache2 stop
root@ecasp:/root> apt-get update
root@ecasp:/root> apt-get install nginx
```

Configurando o NGINX

O arquivo `/etc/nginx/fastcgi_params` deve ficar assim:

```
fastcgi_param  GATEWAY_INTERFACE CGI/1.1;
fastcgi_param  SERVER_SOFTWARE  nginx;
fastcgi_param  QUERY_STRING     $query_string;
fastcgi_param  REQUEST_METHOD   $request_method;
fastcgi_param  CONTENT_TYPE     $content_type;
fastcgi_param  CONTENT_LENGTH   $content_length;
fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_name;
fastcgi_param  SCRIPT_NAME      $fastcgi_script_name;
fastcgi_param  REQUEST_URI      $request_uri;
fastcgi_param  DOCUMENT_URI     $document_uri;
fastcgi_param  DOCUMENT_ROOT    $document_root;
fastcgi_param  SERVER_PROTOCOL  $server_protocol;
fastcgi_param  REMOTE_ADDR      $remote_addr;
fastcgi_param  REMOTE_PORT      $remote_port;
fastcgi_param  SERVER_ADDR      $server_addr;
fastcgi_param  SERVER_PORT      $server_port;
fastcgi_param  SERVER_NAME      $server_name;
```

O arquivo `/etc/nginx/nginx.conf` deve ficar assim:

```

user nobody;
worker_processes 1;

error_log /var/log/nginx/error.log warn;
pid /var/run/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include /etc/nginx/mime.types;
    default_type application/octet-stream;

    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;

    sendfile on;

    keepalive_timeout 65;

    include /etc/nginx/conf.d/*.conf;
}

```

O arquivo ***/etc/nginx/conf.d/default.conf*** deve ficar assim:

```

server {
    listen      80;
    server_name localhost;
    root /home/httpd/html;
    charset iso-8859-1;

    location / {
        index index.html index.htm index.php;
    }

    # desabilita o log do favicon
    location = /favicon.ico {
        log_not_found off;
    }

    # o php funcionando como port
    location ~ /\.php$ {
        # Filter out arbitrary code execution
        location ~ /\.*/.*\.php$ {return 404;}
        include fastcgi_params;
        fastcgi_pass 127.0.0.1:9000;
    }

    # deny para o htaccess
    location ~ /\.ht {
        deny all;
    }

    # sistemas maker, na linha tem que colocar as iniciais do sistema
    # para fazer o redirect
    location ~* /(csp|csp2|edt|probe|rrc) {
        sendfile off;
        proxy_pass      http://127.0.0.1:8080;
        proxy_redirect   default;

        proxy_set_header Host          $host;
        proxy_set_header X-Real-IP      $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_max_temp_file_size 0;

        # configuracao do parametro maximo de upload
        client_max_body_size 10m;
        client_body_buffer_size 128k;

        proxy_connect_timeout 90;
        proxy_send_timeout 90;
        proxy_read_timeout 90;

        proxy_buffer_size 4k;
        proxy_buffers 4 32k;
        proxy_busy_buffers_size 64k;
        proxy_temp_file_write_size 64k;
    }
}

```

Configurando o PHP4

Edite o arquivo **/etc/php4/cgi/php.ini** e modifique a diretriz **cgi.fix_pathinfo** para:

```
cgi.fix_pathinfo=0
```

Agora crie o arquivo **/etc/init.d/php-fcgi** com o seguinte conteúdo:

```

#!/bin/bash
### BEGIN INIT INFO
# Provides:          php-fcgi
# Required-Start:    $nginx
# Required-Stop:     $nginx
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: starts php over fcgi
# Description:       starts php over fcgi
### END INIT INFO

#(( EUID )) && echo "You need to have root privileges." && exit 1
BIND=127.0.0.1:9000
#BIND=/tmp/php.socket
USER=nobody
PHP_FCGI_CHILDREN=15
PHP_FCGI_MAX_REQUESTS=1000

PHP_CGI=/usr/bin/php-cgi
PHP_CGI_NAME=`basename $PHP_CGI`
PHP_CGI_ARGS="- USER=$USER PATH=/usr/bin PHP_FCGI_CHILDREN=$PHP_FCGI_CHILDREN PHP_FCGI_MAX_REQUESTS=
$PHP_FCGI_MAX_REQUESTS $PHP_CGI -b $BIND"
RETVAL=0

start() {
    echo -n "Starting PHP FastCGI: "
    start-stop-daemon --quiet --start --background --chuid "$USER" --exec /usr/bin/env --
$PHP_CGI_ARGS
    RETVAL=$?
    echo "$PHP_CGI_NAME."
}
stop() {
    echo -n "Stopping PHP FastCGI: "
    killall -q -w -u $USER $PHP_CGI
    RETVAL=$?
    echo "$PHP_CGI_NAME."
}

case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo "Usage: php-fastcgi {start|stop|restart}"
        exit 1
        ;;
esac
exit $RETVAL#!/bin/bash
### BEGIN INIT INFO
# Provides:          php-fcgi
# Required-Start:    $nginx
# Required-Stop:     $nginx
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: starts php over fcgi
# Description:       starts php over fcgi
### END INIT INFO

#(( EUID )) && echo "You need to have root privileges." && exit 1
BIND=127.0.0.1:9000
#BIND=/tmp/php.socket
USER=nobody
PHP_FCGI_CHILDREN=15
PHP_FCGI_MAX_REQUESTS=1000

PHP_CGI=/usr/bin/php-cgi
PHP_CGI_NAME=`basename $PHP_CGI`

```

```

PHP_CGI_ARGS="- USER=$USER PATH=/usr/bin PHP_FCGI_CHILDREN=$PHP_FCGI_CHILDREN PHP_FCGI_MAX_REQUESTS=
$PHP_FCGI_MAX_REQUESTS $PHP_CGI -b $BIND"
RETVAL=0

start() {
    echo -n "Starting PHP FastCGI: "
    start-stop-daemon --quiet --start --background --chuid "$USER" --exec /usr/bin/env --
$PHP_CGI_ARGS
    RETVAL=$?
    echo "$PHP_CGI_NAME."
}

stop() {
    echo -n "Stopping PHP FastCGI: "
    killall -q -w -u $USER $PHP_CGI
    RETVAL=$?
    echo "$PHP_CGI_NAME."
}

case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo "Usage: php-fastcgi {start|stop|restart}"
        exit 1
        ;;
esac
exit $RETVAL

```

Adicione o php-fcgi ao start do servidor.

```

root@ecasp:/etc/init.d>chmod +x php-fcgi
root@ecasp:/etc/init.d>inserv php-fcgi
root@ecasp:/etc/init.d>update-rc.d php-fcgi defaults
update-rc.d: using dependency based boot sequencing
root@ecasp:/etc/init.d>

```

Caso esse servidor utilize o **MUNIN** para monitoramento, adicione as linhas abaixo depois do **location ~ /\.ht** do arquivo **/etc/nginx/conf.d/default**

```

# munin
location /munin/static/ {
    alias /etc/munin/static/;
    expires modified +1w;
}
location /munin/ {
    #auth_basic "Restricted";
    # Create the htpasswd file with the htpasswd tool.
    #auth_basic_user_file /etc/nginx/htpasswd;

    alias /var/cache/munin/www/;
    expires modified +310s;
}

```

CONFIGURAÇÃO PARA NGINX e PHP4 usando sockt's

Instale o pacote spawn-fcgi

Configure o arquivo **/etc/nginx/conf.d/default.conf** na diretiva **php** para:

```
# o php funcionando com sockt's
location ~ /\.php$ {
    # Filter out arbitrary code execution
    location ~ /\.*/.*\.php$ {return 404;}
    include fastcgi_params;
    fastcgi_pass unix:/var/run/php-fastcgi/php-fastcgi.socket;
}
```

O modifique o arquivo **/etc/init.d/php-fcgi** para:

```
#!/bin/bash
### BEGIN INIT INFO
# Provides:          php-fcgi
# Required-Start:    $nginx
# Required-Stop:     $nginx
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# Short-Description: starts php over fcgi
# Description:       starts php over fcgi
### END INIT INFO

FASTCGI_DIR=/var/run/php-fastcgi
FASTCGI_USER=nobody
FASTCGI_GROUP=nobody
SOCKET=/var/run/php-fastcgi/php-fastcgi.socket
PIDFILE=/var/run/php-fastcgi/php-fastcgi.pid
CHILDREN=6
PHP=/usr/bin/php4-cgi
PHP_CGI_ARGS="/usr/bin/spawn-fcgi -s $SOCKET -P $PIDFILE -C $CHILDREN -u $FASTCGI_USER -g
$FASTCGI_GROUP -f $PHP"
RETVAL=0

start() {
    [ -d $FASTCGI_DIR ] || { mkdir -p /var/run/php-fastcgi ; chown -R $FASTCGI_USER:$FASTCGI_GROUP
$FASTCGI_DIR; }
    echo -n "Starting PHP FastCGI: "
    start-stop-daemon --quiet --start --background --chuid "$USER" --exec /usr/bin/env --
$PHP_CGI_ARGS
    RETVAL=$?
    echo "$PHP."
}

stop() {
    echo -n "Stopping PHP FastCGI: "
    rm -f $SOCKET $PIDFILE
    killall -q -w $PHP
    RETVAL=$?
    echo "$PHP."
}

case "$1" in
    start)
        start
        ;;
    stop)
        stop
        ;;
    restart)
        stop
        start
        ;;
    *)
        echo "Usage: php-fastcgi {start|stop|restart}"
        exit 1
        ;;
esac
exit $RETVAL
```