Jenkins behind an NGinX reverse proxy

Running Jenkins from a subdomain (like http://jenkins.domain.tld)

Due to people are often struggling getting Jenkins to work behind an NGINX reverse proxy setup I've decided to share my currently running config.

Hope this could be of any help to someone.

```
server {
    listen 80;
    server_name jenkins.domain.tld;
    return 301 https://$host$request_uri;
}

server {
    listen 80;
    server_name jenkins.domain.tld;

    location / {
      proxy_set_header        Host $host:$server_port;
      proxy_set_header        X-Real-IP $remote_addr;
      proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header        X-Forwarded-Proto $scheme;

      # Fix the "It appears that your reverse proxy set up is broken" error.
      proxy_pass          http://127.0.0.1:8080;
      proxy_read_timeout  90;

      proxy_redirect      http://127.0.0.1:8080 https://jenkins.domain.tld;

          # Required for new HTTP-based CLI
      proxy_http_version 1.1;
          proxy_request_buffering off;
      # workaround for https://issues.jenkins-ci.org/browse/JENKINS-45651
      add_header 'X-SSH-Endpoint' 'jenkins.domain.tld:50022' always;
    }
  }
```

Running from a subdomain with SSL

```
upstream jenkins {
  server 127.0.0.1:8080 fail_timeout=0;
}

server {
  listen 80;
  server_name jenkins.domain.tld;
  return 301 https://$host$request_uri;
}

server {
  listen 443 ssl;
  server_name jenkins.domain.tld;

  ssl_certificate /etc/nginx/ssl/server.crt;
  ssl_certificate_key /etc/nginx/ssl/server.key;

  location / {
    proxy_set_header        Host $host:$server_port;
    proxy_set_header        X-Real-IP $remote_addr;
    proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header        X-Forwarded-Proto $scheme;
    proxy_redirect http:// https://;
    proxy_pass              http://jenkins;
        # Required for new HTTP-based CLI
    proxy_http_version 1.1;
        proxy_request_buffering off;
    proxy_buffering off; # Required for HTTP-based CLI to work over SSL
    # workaround for https://issues.jenkins-ci.org/browse/JENKINS-45651
    add_header 'X-SSH-Endpoint' 'jenkins.domain.tld:50022' always;
  }
}
```

Running Jenkins from a folder with TLS encryption (like https://domain.tld/jenkins/)

However, you may want to access Jenkins from a folder on your main web server. This allows you to use the same TLS/SSL certificate as for your top level domain, whereas a sub-domain like *jenkins.domain.tld* may require a new TLS/SSL certificate (that seems to depend on your certificate provider). You can configure nginx as a reverse proxy to translate requests coming in from the WAN as https://domain.tld/jenkins/ to LAN requests to http://10.0.0.100:8080/jenkins.

Note that this example uses the same settings as currently listed on the wiki article at https://wiki.jenkins-ci.org/display/JENKINS/Running+Hudson+behind+Nginx, but with different values for the **proxy_pass** and **proxy_redirect** directives.

```
server {


    # All your server and TLS/certificate settings are up here somewhere
    [...]


    # Nginx configuration specific to Jenkins
    # Note that regex takes precedence, so use of "^~" ensures earlier evaluation
    location ^~ /jenkins/ {

        # Convert inbound WAN requests for https://domain.tld/jenkins/ to
        # local network requests for http://10.0.0.100:8080/jenkins/
        proxy_pass http://10.0.0.100:8080/jenkins/;

            # Rewrite HTTPS requests from WAN to HTTP requests on LAN
        proxy_redirect http:// https://;

        # The following settings from https://wiki.jenkins-ci.org/display/JENKINS/Running+Hudson+behind+Nginx
        sendfile off;

        proxy_set_header   Host             $host:$server_port;
        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;

        # This is the maximum upload size
        client_max_body_size       10m;
        client_body_buffer_size    128k;

        proxy_connect_timeout      90;
        proxy_send_timeout         90;
        proxy_read_timeout         90;

        proxy_temp_file_write_size 64k;

            # Required for new HTTP-based CLI
        proxy_http_version 1.1;
            proxy_request_buffering off;
        proxy_buffering off; # Required for HTTP-based CLI to work over SSL
  }
```

In addition, you must ensure that Jenkins is configured to listen for requests to the /jenkins/ folder (e.g. http://10.0.0.100:8080/jenkins/ instead of http://10.0.0.100:8080/). Do that by adding the parameter **--prefix=/jenkins** to the Jenkins default start-up configuration file. On my system (Ubuntu 12.04 LTS) the configuration file is **/etc/default/jenkins**. For example, here's the full JENKINS_ARG parameter list (the only part I added was **--prefix=/jenkins**):

```
JENKINS_ARGS="--webroot=/var/cache/jenkins/war --httpPort=$HTTP_PORT --ajp13Port=$AJP_PORT --prefix=/jenkins"
```

Once configured, you should also set the URL used by the Jenkins UI at **Jenkins** > **Manage Jenkins** > **Jenkins Location** > **Jenkins URL** to something like: "https://domain.tld/jenkins/.

Running Jenkins behind AWS ELB from a subdomain (like https://jenkins.domain.tld)

This example is suitable for the following request flow: **User AWS ELB nginx Jenkins**

If you don't use SSL on AWS ELB, remove redirection from HTTP to HTTPS in config below.

In case Jenkins is not the only service on your host, replace **server_name localhost** with your actual domain name and remove **default_server**.

```
server {
  listen 80 default_server;
  server_name localhost;

  # Remove nginx and OS versions from server header
  server_tokens off;

  # Redirect HTTP requests to HTTPS basing on X-Forwarded-Proto header from AWS ELB
  if ($http_x_forwarded_proto = 'http') {
    return 301 https://$host$request_uri;
  }

  # Extract real IP from X-Forwarded-For header to see user IP in nginx logs
  set_real_ip_from  10.0.0.0/8;
  set_real_ip_from  172.16.0.0/12;
  real_ip_header    X-Forwarded-For;
  real_ip_recursive on;

  # Proxy all requests to Jenkins
  location / {
    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_set_header        X-Forwarded-Proto $http_x_forwarded_proto;
    proxy_set_header        X-Forwarded-Port  $http_x_forwarded_port;

    proxy_pass              http://localhost:8080;

    # Required for new HTTP-based CLI
    proxy_http_version      1.1;
    proxy_request_buffering off;
  }
}
```

Being compatible with CSRF protection

⚠ This section applies to Jenkins 1.x only. Jenkins 2 uses an nginx-compatible crumb header name by default.

If you enable "Prevent Cross Site Request Forgery exploits" in the **Configure Global Security** page, you'll need special care for Jenkins to work behind a proxy. You'll need to enable the **Enable proxy compatibility** checkbox. And you'll need to add to your nginx configuration the following fragment:

```
http {
  ignore_invalid_headers off;
}
```

This is required because Jenkins uses a custom HTTP header named `.crumb`. See bug https://issues.jenkins-ci.org/browse/JENKINS-12875 for details.