

LinkPulse Logging

High traffic environments: Recommended architecture
and examples for Apache mod_rewrite installation

Technical contact: Bendik Heltne: bendik@onsite.no
Business contact: Håkon Tillier: tillier@onsite.no



Introduction

Basic logging in LinkPulse is logging all clicks on the site in question. A basic click consists of click type, a from page url, a to page url and a category describing what part of the GUI the click was done.

Reports in LinkPulse are real-time, which requires extremely effective logging mechanisms to maintain the applications value to its users. For high traffic sites the logging also needs to be efficient in order to catch all desired data at all.

We know of no faster, reliable or more efficient method of data capture than to utilize the built in functionality in the Apache webserver, which this document is all about.

There are several ways to implement LinkPulse logging using Apache. In this paper we describe three different scenarios, if none of these fits your architecture or policy, there are other ways as well. Feel free to contact us for a discussion or joint development project.

However, all three scenarios in this paper are equal in terms of efficiency.

Apache LinkPulse logging in general

Linkpulse takes advantage of Apaches stability and performance and uses a virtual host containing a rewriterule, custom logformat and a customlog pipe to a perl-program. The perl-program aggregates loglines and flushes when it has reached one of the predefined thresholds for volume or time.

This mechanism also ensures that you can implement logging on servers that otherwise have no write access (e.g. When frontend has SAN or NFS access with read only attributes)

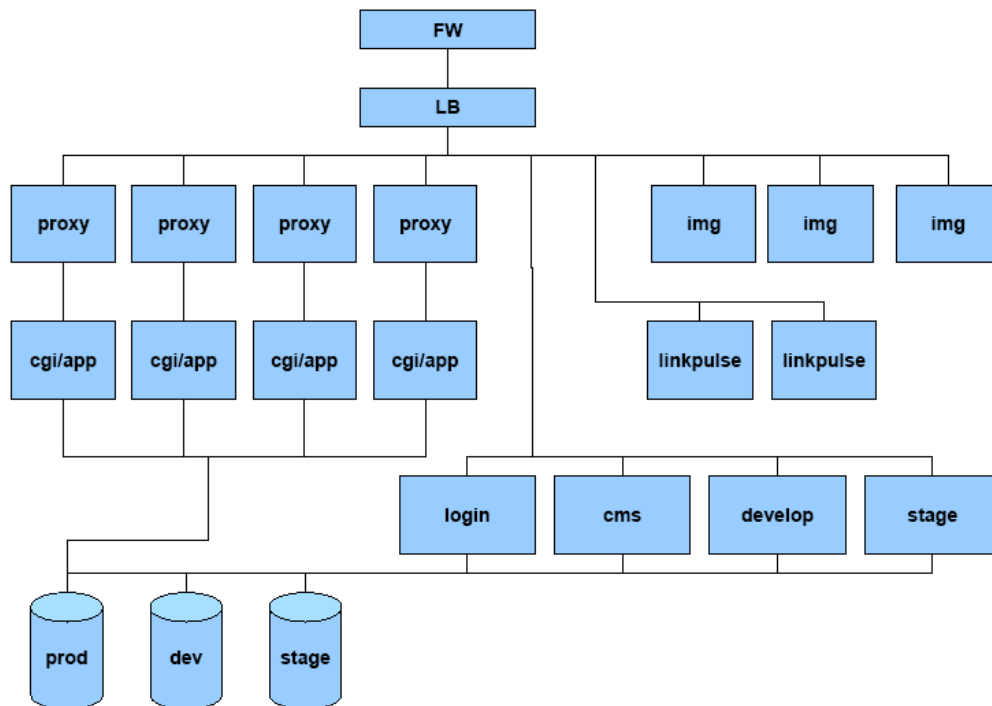
The servers should respond to a http request using an hostname (or passed through by the proxy when host requested matches). Examples for hostname are go.example.com, lp.example.com - or something completely different.

The different scenarios are all about where you put the logging functionality in your rig. There is really no absolute correct answer, but these three are well tested.

1. LinkPulse in a standalone environment

This is the recommended solution in an environment that does not use apache as frontend or backend servers, or where policy gives restriction for multiple role machines.

You use two LinkPulse log servers set up as standalone boxes and placed directly behind the load balancer. The two servers are for redundancy not capacity. These servers should run a standard Linux installation (requirements are Apache 1.3.x or later and Perl 5.6.x or later)

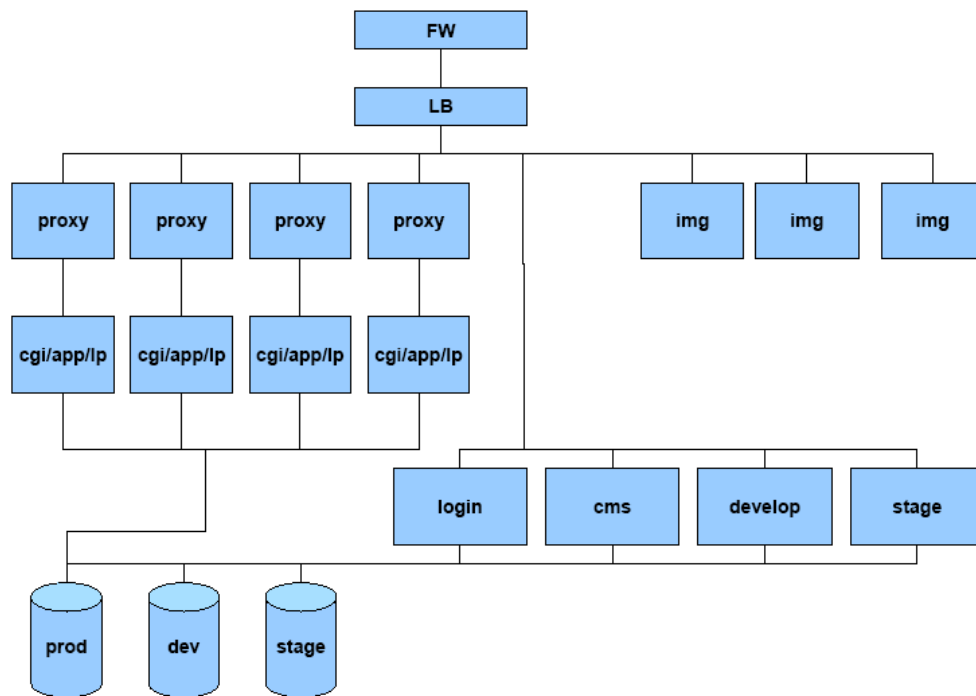


2. Utilizing existing apache backend servers

This is an example on how you could take advantage of existing servers running apache. The logger is so effective that it will not influence serverload. The proxies will receive the requests and pass it through to the linkpulse server.

This requires that the proxy servers must have rules to avoid caching any requests to the LinkPulse logger.

Redundancy is equal to the number of servers serving the web.



3. Utilizing existing apache front-end servers

You could also use the frontend servers which run proxy software to setup apache and use it for logging. Redundancy is equal to example nr. 2 and you will not need to run request trough the proxy.

