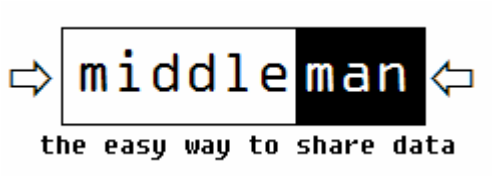


Getting started with middleman

General purpose

The purpose of middleman is to provide a quick and easy way to make available over a corporate intranet (or the Internet) various contents that would be otherwise difficult to access because of the network topology or technologies.

The appliance has the following features:

- On-the-fly web sharing of UNC resources (SMB shares), with access control
- Configuration of reverse proxies
- Creation of persistent web page grabbers
- DNS querying and ping/traceroute

All these features are available from a user-friendly web interface, in real-time. No login is required in the VM shell; the DHCP-provided address is displayed on the logon screen, as well as the default username and password for the web interface.

Audience

The middleman virtual appliance has been designed as a tool for systems administrators and technical support specialists. However, the appliance is easy to use and could be of some interest for SOHO users running a small network. It could also come handy for someone looking for an easy way to publish web content without having to manage a web server.

Usage scenarios

Scenario 1: the happy CEO

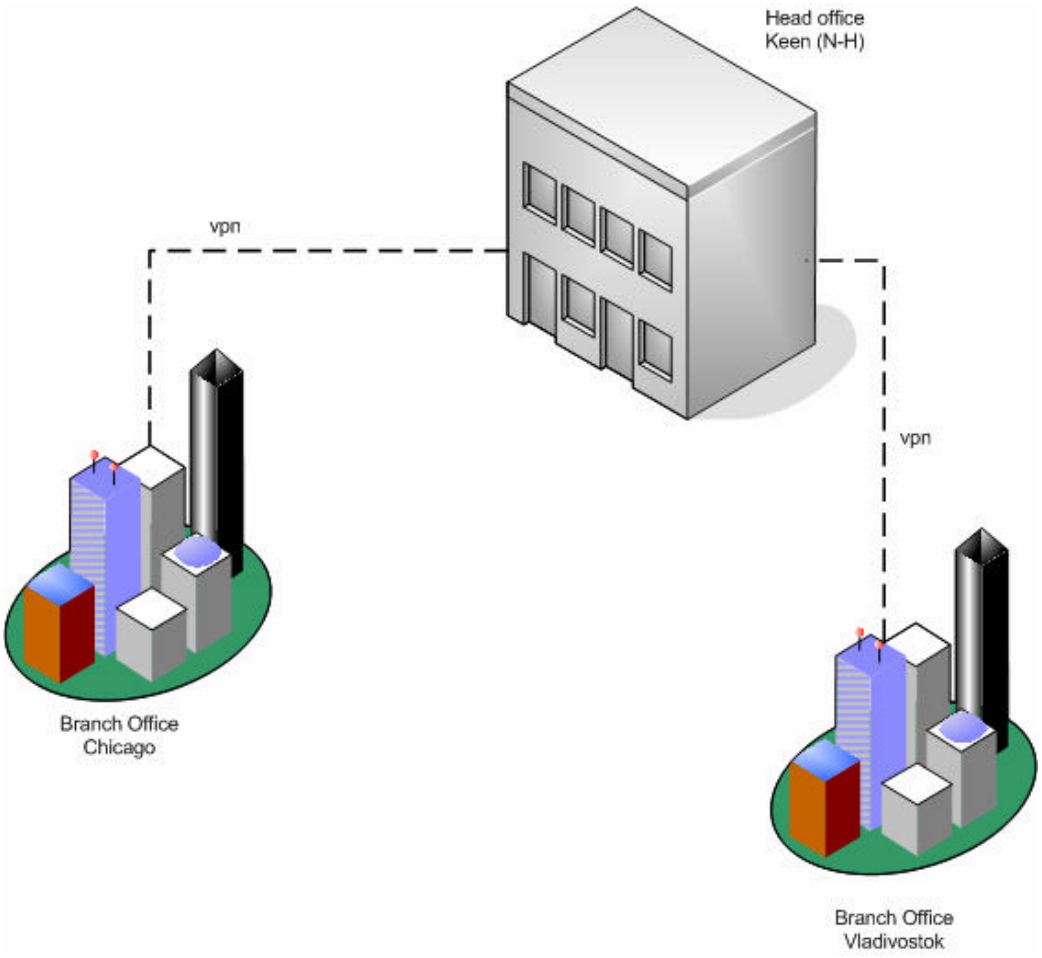
The CEO is in a meeting with an important customer (or maybe a SEC investigator). He calls the helpdesk because he would like to show various reports but he does not know how to access the archived reports from the customer's office.

- **Without middleman:** *the technician tries to find out how to remotely configure the CEO's laptop to connect to a different file server across the VPN, then finally gives up and start sending the reports by email.*

- **With middleman:** *the technician creates a SMB share on the archive server, then links this resource to a new web share in middleman and gives the IP to the CEO, who can browse the folders and download copies of the reports.*

Scenario 2: the branch office concern

The Company has a data center located in Keen (N-H), a branch office in Chicago and another branch office in Vladivostok. Both branch office are connected to the data center via VPN; however, for economical reasons there is no VPN between the two branch offices.



The director in Vladivostok heard about an interesting intranet application at the Chicago office, where results are posted on a daily basis. He would like to have access to this application, but he does not have the budget for a VPN link and the routing at the data center does not allow communication between the two offices.

With middleman, the helpdesk can quickly setup a reverse proxy located in the data center, where the server has access to both offices. Therefore the Vladivostok director can happily access the intranet application (and he is likely to bring a bottle of vodka to the helpdesk at his next visit in Keen).

Scenario 3: the tired sysadmin

Joe is a tired sysadmin who is in charge of four branch offices that are sadly not linked by a fully-meshed network. His only remote access to the sites is with VNC. One morning, he notices that he has no connection to the most remote of all sites.

In the previous weeks he was wise enough to install middleman at every site. So he calls a secretary at the remote site and guides her to the network utilities page of middleman. By launching a trace from middleman to a public website, he finds out that a specific router is not responding, and he can resolve the issue by talking with the local telco, without leaving his comfortable basement office.

Features that could be added to middleman

With little efforts, middleman could be improved by adding the following features:

- On-the-fly FTP and SFTP virtual directory
- Complete URL rewriting instead of basic reverse proxy
- Access control on reverse proxies and page grabbers

System requirements

Middleman has very little requirements:

- 32 MB of RAM (could be increased to support heavy traffic)
- 250MB virtual disk (which is actually less than half full to allow plenty of logs or dynamic content)
- A network card
- Access to a DHCP server (or a sysadmin able to pick a static address)

Technical specifications

- O/S: Slackware Linux 10.2
- Kernel 2.6.16 (older kernel available in the boot folder)
- Partition scheme: one single 250MB partition (plus a potential 30MB swap partition that can be enabled in fstab)
- Actual size of the content: 114MB
- The appliance relies heavily on bash, Apache, PHP and of course Samba
- Default root password: “password” (without the quotes)
- Web technologies: Apache 2.0 & PHP4
- No database (file-based configuration)
- Default web user: admin
- Default web password: admin
- VMWare tools not installed

middleman could have been smaller but in order to facilitate the customization of the appliance, all Apache modules have been included, as well as the VIM editor.

Overall architecture and technical details

- All interaction between the user and the appliance is made from a web interface (PHP/Apache)
- Many of the actions are actually executed by PHP or Bash via a system call (low-privilege)
- For the most critical items, the web application puts a flag in a folder that is monitored by a root-launched application (/root/listen)
- The root-launched application executes various scripts to mount or umount resources, delete files, and so forth. Root scripts are located in /root.
- All web pages are stored under /var/www (admin for the main application, grabber for the web page grabbers). Non-root scripts are located in /opt/scripts.
- Various alias and proxy configuration files are created by the web application and taken in account by Apache (with a kill –HUP launched by the root application). Those scripts are located under /opt/scripts
- Users and groups are stored in /var/www/config. The standard htpasswd is used to edit the accounts.
- Apache and PHP are located under /opt
- The bootloader is LILO but no prompt is showed. A recent kernel is launched by default but an older one is available (commented out in lilo.conf).
- The primary concern of the architecture is the KISS paradigm...

About the author

middleman has been designed by Jean-Luc Martel, a systems administrator working in Montreal (Canada). Email: lucm@iqato.com