Paper Title

Structure and Performance of Open Access Networks – Case Lappeenranta Model

M.Juutilainen, T.Lapinlampi, J.Ikonen and J.Porras

Laboratory of Communications Engineering, Lappeenranta University of Technology, Lappeenranta, Finland
e-mail: matti.juutilainen@lut.fi

Abstract Title & Keywords Title

Authors & Affiliations

Abstract

Abstract

Lappeenranta Model is a framework for building open operator neutral access networks with local services. The focus is in locality: to provide an easy local network connection available to everyone and to encourage people on developing their knowledge and abilities on networking. This publication describes the operating principle, structure and different implementation options and analyzes the performance of Lappeenranta Model.

Keywords

Abstract Title & Keywords Title

Normal

Lappeenranta Model, Open Access Network, Operator Neutral Network

1. Introduction

Heading 1

Normal

The trend in modern WLAN networks is changing from closed single-ISP networks towards inter-network roaming and public access systems that allow multiple ISPs, service providers and users to share the same access medium. These open access networks bring savings for the network builder and enables better service level and lower connection fees for end users.

Open access networks differ from traditional closed network in the sense that ISPs can no longer reach monopolistic position in the market, which reduces the overall service level and raises the costs for end users. Open access will most likely change the networking future as it will bring new technical, business and service opportunities as reaching the end users is no longer as problematic and expensive as it has been.

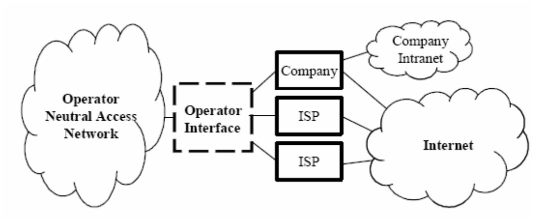
In this paper, we concentrate on one example of open access networks: Lappeenranta Model. We will describe the main concepts, operating principle, structure and different implementation options of Lappeenranta Model and evaluate the model's performance.

2. Lappeenranta Model

Heading 1

Lappeenranta model is an open access network model allowing end users and service providers to connect to a shared local access network. Lappeenranta model opens the access network to everyone and can therefore be compared to community networks, like Seattle Wireless.

The heart of Lappeenranta Model is Operator Interface. It is a cluster of servers located between an operator neutral (wireless) access network and ISPs providing the connections onwards to other networks (see Figure 1). Every connection between the access network and the ISPs is supposed to travel through Operator Interface, which can be therefore compared to Access Controllers. However, unlike normal Access Controllers, Operator Interface has support to multi-ISP environment and multiple additional features, which will be described more detailed later in section 2.2.



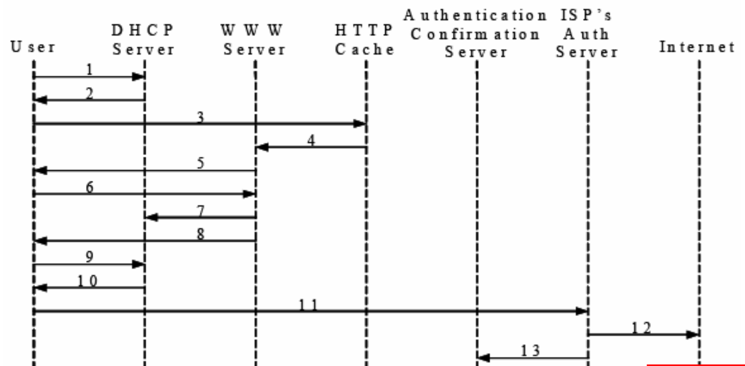
Figure/Table Caption

Figure 1: Basic structure of Lappeenranta Model

2.1. Operating Principle

Heading 2

Let us assume that a new user (i.e. a user that has not connected to the network before) enters the coverage area of the open access network. The user has a device with a network card (for example a WLAN card) connected and operating correctly. The user wants to access some web page in the Internet and starts a web browser to retrieve the page. Figure 2 illustrates what happens in the network.



Figure/Table Caption

Figure 2: Flow chart describing messages between Operator Interface's components

1. In order to work properly in the access network user's device needs an IP address. It sends a request for a new IP address.

2. DHCP server hears the request and checks from the user database if the device is already known. In this case the user is new and the MAC address of the device cannot be found from the database. The DHCP server gives

the device an IP address belonging to an address block of unregistered users. These temporary IP addresses have a short lease time (for example one minute) in order to enable a registration process without long delays. Access network's IP addresses can be either private or public, but private addresses seem more suitable as every device in the access network needs an IP address and nowadays it can be difficult to get large number of public addresses.

3. Having the new temporary IP address, the device sends a request for a web page asked by its user. Let us assume that the user wants to browse in Internet.

4. Because the user has not yet been registered in the network (i.e. the device has a temporary IP address), HTTP cache catches the request for the web page and forwards it to Operator Interface's WWW server. Devices having these temporary IP addresses can only get to a login page and some other predefined pages (for example information on the local network, user instructions etc.). Every user has to register in order to get access to the actual content of the network or to the ISPs providing connections to Internet.

5. The WWW server sends a login page to the user. The login page includes a selection of the ISPs that can be used. There is also an option not to choose ISP at all. Without ISP, the user can freely browse the access network content and any whitelisted addresses from Internet. However, an ISP is required for unrestricted Internet connection.

6. After the user has selected the ISP, the device sends the information from the web page form to the WWW server.

7. A script in WWW server processes the ISP choice and informs the DHCP server to change the device's IP address. The user database is updated and the DHCP server is restarted to refresh the new IP configuration.

8. The WWW server also sends a web page to the device informing the user to wait a moment until the IP address update is complete.

9. As the lease time of the temporary IP address expires, the device requests a new IP address from the DHCP server.

10. The DHCP server now finds the device's MAC address from the user database and gives a new IP address according to the chosen ISP. This will happen automatically in subsequent connections. Each of the ISPs has its own predefined IP address space.

11. Now the device has an IP address that will be routed outside the access network if requested. The user has asked a WWW page from the Internet, so the request is routed to the ISP chosen by the user. The ISP is responsible for authenticating users trying to connect Internet through ISPs connections.

12. ISP’s authentication server authenticates the user before allowing a connection to the Internet. Authentication can be done by any means and the ISP can use its existing authentication mechanism (login name & password, Radius, electronic certificate, etc.) After a successful authentication procedure, the authentication server will allow the user to access the Internet. The subsequent connections will automatically be routed through the ISP's connections (until the lease expires).

13. If required, the ISP’s authentication server may also send a confirmation message to an optional Authentication Confirmation Server located in Operator Interface. The confirmation message consists of information whether the authentication procedure was successful or not. The message may also include additional information such as the expiration date of the authentication or the user's credential information. The message can also be used to block suspicious users from connecting to the ISP anymore, if needed.

It was mentioned in previous list under item 2 that one reason for using private IP addresses is the lack of free addresses. Another reason is that when using private addresses, a large amount of devices in the access network can easily be addressed in the same network so that they can hear each other. This allows direct local communications and prevents the need for circulating local traffic through Internet service providers and Internet. When connecting to Internet through some of the ISPs, the private address can be translated to public addresses.

2.2. The Structure

Heading 2

Lappeenranta Model is based on computers running Linux operating system. Each of the computers hosts some services required to operate the network system. As section 2.3 later describes, there are different options for implementation, but let us first describe the components and the operating principle of the system.

The core of Lappeenranta model is called Operator Interface. Operator Interface includes all the components needed to run Lappeenranta model and acts like an Access Controller extended with several additional features. Figure 3 shows an example distribution of the components of Operator Interface. There are three main operational components: Access Controllers, Other Common Services (OCS) and Name Service.

Access Controllers are responsible for routing access network users to correct ISPs. An HTTP cache connected to user routing is used for redirecting users to a login page when necessary and providing network users with pushed announcements or advertisements. Each Access Controller can handle one or multiple ISPs and can be physically located wherever in the access network. Redirector is used for checking from the Database whether to show a user another page than requested. For example, the user can be advised to use DHCP, show different advertisements, announcements or login pages. The transparent HTTP cache is implemented with Linux iptables tools redirecting all the traffic from port 80 to local port listened by Squid.

Other Common Services include the main functions for managing the access network: login pages to users for selecting ISPs, DHCP services, and Radius services for authenticating devices in the Access Points of the access network. Name Service is used for providing the access network with alphabetical domain names that are easier to remember than IP addresses.

Name Service includes DNS (Domain Name Service) and NTP (Network Time Protocol). Database is for storing information from DHCP service and from network's Access Points.

Although there are many components, only three of them are crucial for the operation of Lappeenranta model. These components are Redirector, HTTP Cache and Database. In principle, everything else is optional, but highly recommended in order to maintain all the functionality and to enable easy connectivity.

Please note that Figure 3 shows only a logical component view of Lappeenranta Model. The figure does not commit on the actual locations of the components, as they can be installed on different servers or locations like described later in section 2.3.

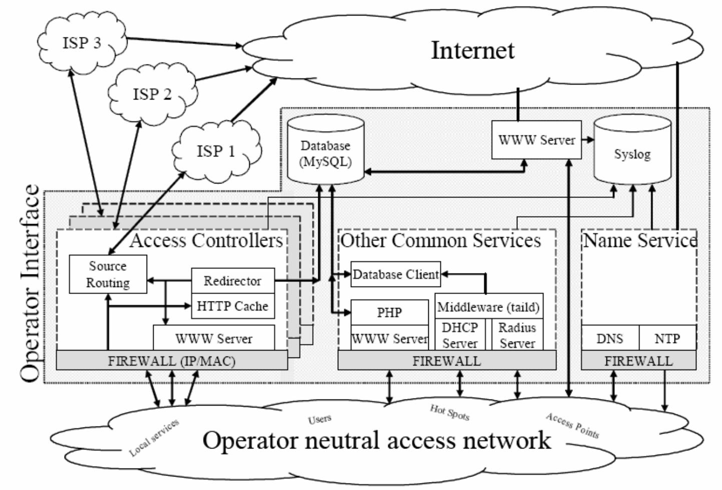


Figure 3: Lappeenranta Model – A logical component view

Some of the components shown in Figure 3 are required for the basic operation of the system; some of them may be implemented for additional features. See Table 1 for more detailed descriptions and necessity of the components.

Component	Necessity	Description
Redirector	Required	Redirector in co-operation with source-address-based routing is required in order to route traffic correctly between users and correct ISP. Routing is implemented by using an iptables firewall, which matches IP and MAC addresses. All HTTP traffic is routed to HTTP cache.
HTTP Cache	Required	HTTP cache is required to redirect an unregistered user to login pages located in WWW server. HTTP cache can also be used to force announcements or advertisements to network users. HTTP cache can be used to improve the performance.
User Database	Required	User database is used for storing user information, such as IP and MAC addresses, and optional positioning or authentication information.
DHCP Server	Recommended	DHCP server is necessary for access network administration. DHCP server gives IP addresses to public access network's devices.
WWW Servers	Recommended	WWW servers are used for storing login pages for unregistered network users and for any local network pages.
DNS Server	Recommended	DNS server is needed in order to allow user-friendly host names in the access network. The DNS server is used mainly for local content as most users will use ISPs' DNS servers, which are slaves to the local DNS server.
Log Databases	Recommended	Log databases are used for storing information on network usage and system operation.
Radius Server	Recommended	Radius server is used for authenticating network users. Any authentication method can be used if needed. However, notice that usually there is no need to authenticate users directly in the local access network. The authentication done by the ISPs connecting the users to any external networks (such as Internet).
Whitelist/ Blacklist	Optional	There can be some "free" web addresses (such as local network's home pages or some sponsored services, like online banking services) that are available for everyone in the access network. Every user can access these whitelisted addresses without a connection through an ISP.
Authentication Confirmation Service	Optional	Authentication Confirmation Service (ACS) can be used for collecting authentication information of users already authenticated by ISPs. Services in the local access network, for example, can differentiate their services to end users by using the information from ACS.

Table 1: Components of Lappeenranta Model

Figure/Table Caption

2.3. Different Implementation Options

Operator Interface can be implemented in numerous different forms depending on the requirements. The chosen implementation method is mostly dependent on general environment and network size. Implementation guidelines are different in large-scale and small-scale environments. Cities or large organizations, for example, have different user numbers and network loads than small environments, such as airports or apartment house communities.

Although Operator Interface consists of numerous components, only the required three core components produce high load to the system: redirector (source-address-based user routing), HTTP cache, and user database. These core components can be distributed in different ways to separate server computers to improve the overall performance. The distribution of other components is not critical to the overall performance.

There are three basic options for implementation of Operator Interface:

1. Centralized implementation,

2. Partially distributed implementation, and

3. Fully distributed implementation.

1. Centralized implementation
The first basic option for implementing Operator Interface is to concentrate all the components into one server computer. This single server then has all the functionality and services required to run Lappeenranta Model.

Centralized Operator Interface suits especially well in small environments, like airports, hotels and apartment buildings, where a single computer can handle the load. If needed, some of the services and functions can still be distributed into different computers to increase performance. It may also be possible to implement the centralized Operator Interface using several identical servers balancing the load and providing redundancy. However, this option still needs to be verified.

2. Partially distributed implementation
Partially distributed implementation is a reasonable option for most cases. Partial distribution means that the three main components are distributed into two computers. This means that one of the components is separated from the other. So there are three options: to distribute either user database, user routing, or HTTP cache. The rest of the components can then be installed wherever wanted as they are not critical to the overall performance.

So it is a good idea to keep user routing and HTTP cache together and dedicate another computer for the user database. This will increase the performance compared to the centralized implementation as the database generates quite heavy load to the system. Additionally, distributing the user database to an external computer will increase the security of the system, as the database computer does not have to be connected to the access network.

3. Fully distributed implementation
Fully distributed implementation means that the three high-load components (user routing, HTTP cache and user database) are all distributed in different computers. The other components can be distributed freely, as they produce virtually no load to

the system. Dedicating a separate computer for each of the components is generally good for performance but requires several computers.

Implementation and configuration procedure of a fully distributed system is rather straightforward as each of the components is located on its own computer. On the other hand, keeping the distributed Operator Interface up-to-date requires more work, as every server computer has to be maintained individually.

Separating HTTP cache and user routing from each other brings out the same performance and security problems described earlier in partially distributed implementation option. This means that partially distributed implementation is the reasonable choice for most cases.

2.4. Performance

In order to know more about the performance and scalability of the Lappeenranta model we needed to test it. As mentioned earlier, the most important components for the performance of the Operator Interface are the HTTP cache, redirector and the database. We ran the tests with partially distributed implementation option, where HTTP cache and redirector are located in one server computer and the database is distributed to another computer. We decided to evaluate the system with Web Polygraph, a performance benchmarking tool for caching proxies, origin server accelerators, L4/7 switches, content filters, and other Web intermediaries (Web Polygraph Web Site, 2006).

In the tests, Operator Interface was located between two test networks that simulated the local access network and Internet. In the first test network (local access network where the Polygraph client is located) there was a group of clients acting as normal network users generating requests for web pages, IP address etc. In the second test network (Internet), there were servers returning the requested web pages and other content.

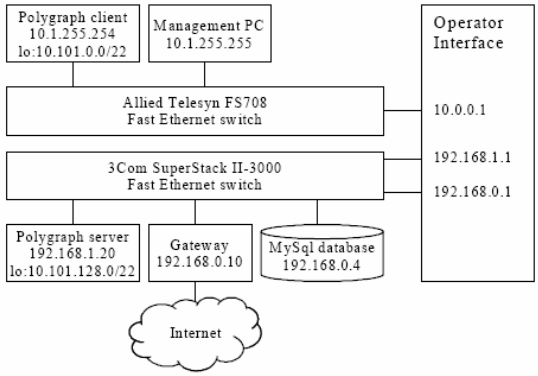


Figure 4: Operator Interface testing environment

As defined by the Web Polygraph test suite "Polymix-4", both the client and server PCs were configured with 1000 local IP aliases on the localhost interface. Traffic to these IP's was routed through the real network interfaces.

Before running the Web Polygraph test suite, the performance of the test network was verified with Netperf. The maximum throughput for tcp streams was 87 Mbit/s, which was reasonably near the maximum theoretical capacity of the network. We assumed that the small slowdown was mainly caused by the oldish Ethernet switch. The maximum speed of the network was not critical for the tests since the bandwidth needed by the test sets was less than 10 Mbit/s. The target HTTP load was 150 requests per second as this was the maximum performance level we could reach with the equipment used.

Operator Interface was configured to route traffic between the Web Polygraph test subnetworks 10.101.0.0/22 and 10.101.128.0/22. Operator Interface acted as a transparent proxy, i.e. it redirected all TCP traffic with destination port 80 to Squid proxy listening at localhost. Web Polygraph was configured to use the test set Polymix-4.

The purpose of the tests was to find out how the redirector affects the overall performance of the web proxy. The performance was expected to decrease, since executing several SQL transactions for each HTTP request takes time.

The tests were conducted in four different setups where used disk system and cache operation were varied:

- with redirector, cache on IDE disk
- with redirector, cache on SCSI disk
- without redirector, cache on IDE disk
- without redirector, cache on SCSI disk

When using SCSI disk, normal delay time under heavy load without using redirector is around 150 ms. With redirector, the delay time rises a bit but stays well below 200 ms. Similarly, with IDE disk, normal delay time is just below 600 ms and with redirector it rises to 800 ms.

As described above, turning redirector on rises the delay times roughly 20-30 per cent under heavy load. It seems that Squid's response times are much more dependent on the performance of the cache disk than any other factor in this test. Compared to SCSI disk, using IDE disk quadrupled the delay times. Having decent disk IO performance is critical for high load environments, but it is still feasible to implement small scale services utilizing IDE disks.

All the solution-specific functionality in Operator Interface (for example pushed advertisements, new logging users etc.) could not be tested due to the limitations of Web Polygraph. We estimate that usage of these functions would not have noticeable affect to the final results as the three main components (user routing, HTTP cache and user database) produce practically all the load to the system.

3. Conclusions and the Future

Lappeenranta Model is technically quite challenging open access model. Unlike other open access solutions (described for example in (Juutilainen et al., 2004)) it allows direct access to local services. The solution is likely to boost technical development of local people, generate new business and allow new Information Society service concepts. The model suits well for different environments, such as airports, campuses and cities. It can be moved and adapted to meet different demands. The system is built by using Linux and public GNU GPL licensing.

In the future, Lappeenranta Model has to be further tested in different environments. Currently there is a test network: Wireless Lappeenranta network, WLPR.NET (Wireless Lappeenranta Network project Web Site, 2006) in the city of Lappeenranta, Finland, where the system is under development. The network is built mainly using WLAN access points connected to Ethernet core network. The model itself allows using any network technologies as long as the network traffic can be routed with each other.

4. References

Heading 1

Battiti, R., Cigno, R.L., Orava, F. and Pehrson, B. (2005), “Wireless LANs: from War Chalking and to Open Access Networks”, *Mobile Networks and Applications*, Volume 10, Number 3, June 2005, Springer Science & Business Media B.V, pp275-287, ISSN: 1383-469X (Paper) 1572-8153 (Online).

Juutilainen, M., Ikonen, J. and Porras, J. (2004), “Comparison of Different WLAN Network Models”, *Proceedings of the 3rd IEEE International Conference on Networking*, Gosier, Guadeloupe, French Caribbean, 2004, pp. 374-381.

StockholmOpen.net Project Web Site (2006), http://www.stockholmopen.n[illegible] (Accessed 28 March 2006)

Web Polygraph Web Site (2006), http://www.web-polygraph.org/. (Accessed [illegible] March 2006)

Wireless Lappeenranta Network Project Web Site (2006), http://www.wlpr.[illegible] Accessed 28 March 2006)

References List