

ECE/CS 4984: Wireless Networks and Mobile Systems
Pre-lab and In-class Exercise 12 (L12)

Part I – Objectives and Lab Materials

Objective:

The objectives of this lab are to:

- ☐ Understand the role of service discovery in pervasive computing
- ☐ Familiar with Service Location Protocol (SLP)

After completing the assignment, you should be able to:

- ☐ Understand how to utilize SLP and Web Services for the development of Web service based pervasive computing applications

Hardware to be used in this lab assignment:

- ☐ Dell notebook with 802.11b card (with a fully charged battery)
- ☐ iPAQ with 802.11b card and cradle (with a fully charged battery)

Software to be used in this lab assignment:

- ☐ OS: Windows 2000 and iPAQ Pocket PC 2002
- ☐ PocketTV and PocketTV SDK
- ☐ mSLP for Service Location Protocol (SLP) implementation

Part II – Pre-lab Assignment

This portion of the assignment should be completed *prior* to the in-class lab session.

Reading Assignment:

- ☐ Service discovery in the future for mobile commerce, by Dipanjan Chakraborty and Harry Chen at <http://www.acm.org/crossroads/xrds7-2/service.html>
- ☐ Documentation on SLP (with mesh enhancement) implementation from Columbia University <http://www.cs.columbia.edu/~zwb/project/slp>

Tasks:

You are expected to perform the following tasks:

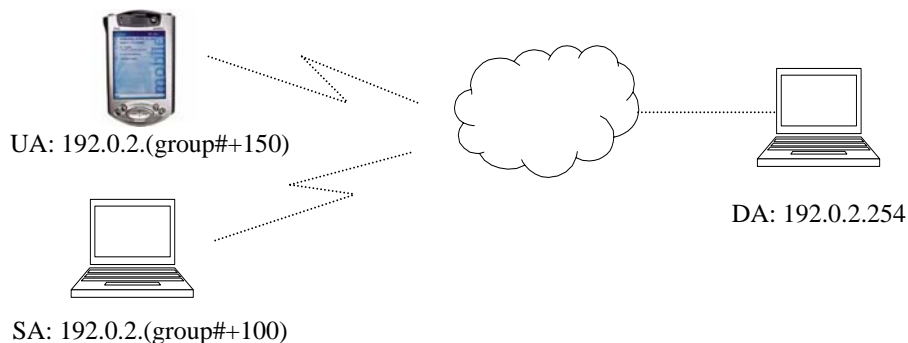
1. Verify the web server (IIS) runs well on your notebook
2. Download mslp-1.0.9.zip from the class web site into your notebook's c:\wmsd\labs\lab_13\mslp\src folder.

3. Download the User Agent (UA) program, mplayer.exe from the class web site, and copy it to your iPAQ.
4. Download at least two video clips from <http://www.pocketmovies.net/> to your IIS web server directory (c:\inetpub\wwwroot by default). As you will register these video clips with a Directory Agent (DA) program, these video clips should be accessible via their URLs. For example, if you download a video file "video1.mpeg" to your web server directory, the video file should be accessible from your browser at <http://localhost/video1.mpeg>.

Part III – In-class lab assignment

You are expected to perform the following tasks:

1. Setup wireless network.
 - o Two Intel wireless gateways will be setup by the GTA
 - i. SSID1 = **ECECS4984A** SSID2 = **ECECS4984B**
 - ii. WEP enabled with key1 = **ABCDE4984A** key2 = **ABCDE4984B**
 - o Two DAs will be running on GTA's notebooks. DA's IP address is 192.0.2.254.
 - o **Groups #1-7 use SSID1, WEP key1 and DA1 and all other groups use SSID2, WEP key2 and DA2.**
 - o Setup your notebook as a SA (notebook) with the IP address 192.0.2.(100+group#)/255.255.255.0
 - o Setup your iPAQ as a UA (iPAQ) with the IP address 192.0.2.(150+group#)/255.255.255.0
 - o Verify your notebook can communicate with your iPAQ and DA.



2. Build a SA based on mSLP (an implementation of SLP using Java) using JDK 1.3.1 and run it on your notebook to register your video services with the DA.
 - ❑ Go to Start->Run and enter "cmd" to open a command line window.
 - ❑ Change directory to c:\wmsd\labs\lab_13\mslp\src.
 - ❑ Build a SA by typing the following command: "c:\jdk1.3.1\bin\javac *.java" followed by <RET>.
 - ❑ Start the service agent by typing the following command "java sa" followed by <RET>. This service agent will register your video services using mSLP.

- ❑ Register your video clips with a DA:
 - a. Click the **Auto DA Discovery** button to search for DAs. The SA application will send a SrvRqst (searching for directory-agent) message to the SLP multicast address 239.255.255.253 at port 2427 (Note: 427 is reserved by SLP v2. You need to have the root privilege to use this port. For demo purposes, we choose to use a port number higher than 1000).
 - The DA running on the GTA's notebook at 192.0.2.254 should be found.
 - b. Register your video clips with the DA. On the **Message Composer** Panel, choose **Message Type** Service Registration (SrvReg) and fill out the required fields:
 - Scope: A scope is used to group services by location, network or administrative categories for increasing SLP scalability. Enter "default" as the scope.
 - Service type: each service has a service type. For example, the service types for <http://www.srvloc.org> and service:lpr://thebe.nvc.cs.vt.edu are "http" and "service:lpr"(printing service), respectively. Enter "video" as the service type.
 - URL: This is the service address. It is unique for all services. Enter the URL to access your video clips residing on you notebook, for example: http://your_ip_address/video1.mpeg.
 - Lifetime: A service lifetime is used to specify the registration period. A registration needs to be refreshed before its lifetime expires; otherwise, the service will be removed from the DA.
 - Language tag: Enter "en" for English.
 - c. Register your video clip with its properties (e.g. URL, attribute) specified by clicking **SendMessage**. Make sure your message composer panel looks similar to the following before you register each video clip:

Message Composer	
Message Type:	Service Registration (SrvReg)
Scope:	default
Service Type:	video
URL:	http://192.0.2.100/video1.mpeg
Life Time:	65535
Language Tag:	en
Attribute:	
<div> <div>Clear All Fields</div> <div>Load Default Values</div> <div>Send Message</div> </div>	

- d. For service registration, you can choose to use **UDP** or **TCP** and **Fresh SrvReg** or **Incremental SrvReg** from the **Configuration** menu. Select UDP and Fresh SrvReg.
 - e. You can start Ethereal to trace packets for better understanding the SLP operation.
- 3. Deploy "mplayer" application onto your iPAQ.
 - ❑ The mplayer application running on your iPAQ serves as a UA to locate video clips services through a DA and plays the video files selected. Run the mplayer application from iPAQ. On the UI interface, specify the maximum video playback rate and click the **Search** button to look for video services.

- mplayer will first send a SrvRqst message (searching for the directory-agent) via the SLP multicast address 239.255.255.253 to locate a DA. If successful, it will send a unicast SrvRqst UDP packet to the DA to locate video clips services that satisfy the specified bandwidth requirement. The source code for locating services operation is listed as follows:

```

struct sockaddr_in broadcast_addr;
struct sockaddr_in peer_addr;
int addr_len = sizeof(peer_addr);
SOCKET sock;
/* Create the Socket */
if ((sock = socket(AF_INET, SOCK_DGRAM, IPPROTO_UDP)) == INVALID_SOCKET) {
    MessageBox(NULL, TEXT("Unable to create Socket "), TEXT(""), MB_OK);
    return;
}

//set broadcast option
BOOL fBroadcast = TRUE;
if (setsockopt ( sock, SOL_SOCKET, SO_BROADCAST, (CHAR *) &
    fBroadcast, sizeof ( BOOL ))==SOCKET_ERROR) {
    MessageBox(NULL, TEXT("Unable to set broadcast mode."), TEXT(""), MB_OK);
    return;
}

// Try Using Sendto instead of send
ZeroMemory((char *)&broadcast_addr, sizeof(broadcast_addr));
broadcast_addr.sin_addr.s_addr = inet_addr ( mcast_addr );
broadcast_addr.sin_family = AF_INET;
broadcast_addr.sin_port = htons(port);

char packet[1024];

/* search for DA :multicast datagram*/
int xid = GetTickCount() % 10000; //random message id
memset(packet, 0, sizeof(packet));
/* compose a SrvRqst message to locate DA */
int msglen = ComposeSrvRqst(packet, xid, mcast_flag, "en", "", "service:directory-agent", "", "", "");
sendto(sock, packet, msglen, 0, (sockaddr *)&broadcast_addr, sizeof (broadcast_addr));
int bytes = recvfrom(sock, packet, sizeof(packet), 0, (sockaddr *)&peer_addr, &addr_len);
/* Process the reply message */
HandleReply(hDlg, packet, xid);

/* send unicast SrvRqst message to DA */
xid++;
memset(packet, 0, sizeof(packet));
msglen = ComposeSrvRqst(packet, xid, normal_flag, "en", "", "video", "default", "bw<=Your_Input", "");
/* DA_Ipaddress is got from the previous message reply */
broadcast_addr.sin_addr.s_addr = inet_addr ( DA_IPaddress );
sendto(sock, packet, msglen, 0, (sockaddr *)&broadcast_addr, sizeof (broadcast_addr));
bytes = recvfrom(sock, packet, sizeof(packet), 0, (sockaddr *)&peer_addr, &addr_len);
HandleReply(hDlg, packet, xid);

closesocket(sock);

```

- The matched services will be listed in a list-box UI from which you can choose one video clip to play.

- You can start Ethereal to trace packets for better understanding of the service discovery and delivery operation.
- Close mplayer on iPAQ by Tools->Exit since the close button on the UI does **NOT** close mplayer completely.