# ISE Guest Password Integration with SMS Gateway Based on Postfix and Kannel Configuration Example

**Document ID: 116918**

Contributed by Michal Garcarz, Cisco TAC Engineer.
Dec 23, 2013

## Contents



## Introduction

This document describes how to integrate open source solutions (Postfix, Maildrop, Kannel) with the Cisco Identity Services Engine (ISE) in order to deliver a Short Message Service (SMS) message to users with guest accounts.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of these topics:

- Cisco ISE and Guest Access
- Linux and Shell Scripting

## Components Used

The information in this document is based on these software and hardware versions:

- Cisco ISE Version 1.2 or later
- Postfix Version 2.10
- Maildrop Version 2.6.0
- Kannel Version 1.5.0

***Note***: Please be informed that Postfix, Maildrop, and Kannel are open source solutions, and Cisco does not support these products. This configuration example simply presents how ISE can be integrated with other products in order to deliver an end−to−end solution.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

# Background Information

ISE allows you to create guest accounts for temporary network access, typically for guests, visitors, contractors, consultants, and customers. Such accounts are created by sponsor users via the Sponsor Portal. When you create the account, it is possible to send a dynamically−generated access password with an SMS directly to the guest user mobile phone.

Cisco ISE is able to send these credentials via email with Simple Mail Transfer Protocol (SMTP) to the Mail2SMS gateway. This gateway is responsible for SMS delivery.

# Configure

## Benefits of the SMS Gateway

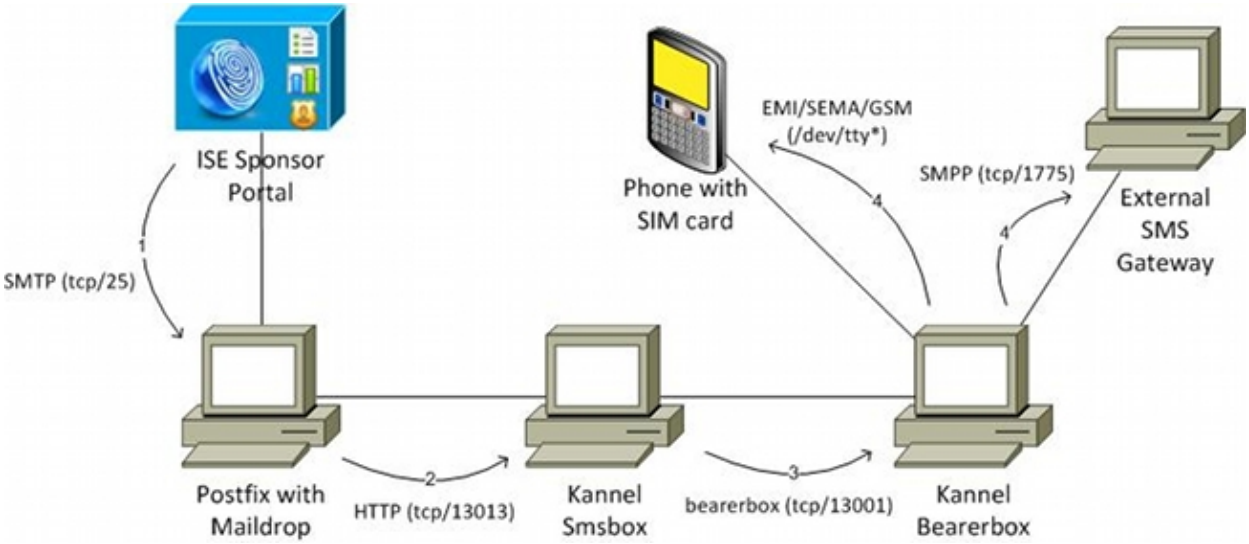
There are multiple Mail2SMS gateway solutions on the market. They can usually receive data with the use of different protocols, such as SMTP, Short Message Peer−to−Peer (SMPP), FTP, HTTP (Simple Object Access Protocol (SOAP), web services), and send an SMS message to the specific mobile phone.

It might be best to build your own SMS gateway. It allows for:

- Greater flexibility
- The ability to build compound rules about routing (time−based, policy−based, content−based)
- Integration with local databases (for example, different routing policies for different Active Directory groups)
- Potentially lower operational costs (no need to pay for an external service)
- The possibility to use this solution also for health alerts generated by ISE and sent as emails

It might be worthwhile to have a mixed deployment − a personal SMS gateway that is also integrated with an external service.

## Network Diagram and Traffic Flow




Here is the flow:

1. The sponsor user creates a guest account with an SMS notification, and provides the mobile phone number for the user. ISE sends an email to the configured SMTP server. The source address (***From***) belongs to a specific sponsor user, whereas the destination address (***To***) is configured globally on ISE (in this example, ***sms@test−cisco.com***). All of the details about the newly created user, such as the username and password, are inside the body of that email.

2. The email arrives on the Postfix server, which is configured with maildrop as a local delivery agent. Just before delivery to the SMS user maildir directory, maildrop searches for mailfilter in the home directory for the user. The mailfilter script parses the email, and if all of the necessary data is found, it uses ***wget*** in order to send the HTTP GET request to the Kannel smsbox. That HTTP GET requests contains the text message along with the username and password, and the mobile phone number of the user. Kannel smsbox is the front end of Kannel that is used in order to accept all requests from users in order to send SMS (to pass it to the Kannel bearerbox).

3. The Kannel smsbox sends that request to the Kannel bearerbox, which has the responsibility to send the SMS.

4. There might be multiple rules and Short Message Service Centers (SMSCs) configured on the bearerbox. This example uses an external SMPP server. Configuration for a locally−attached mobile phone is easy and is presented later.

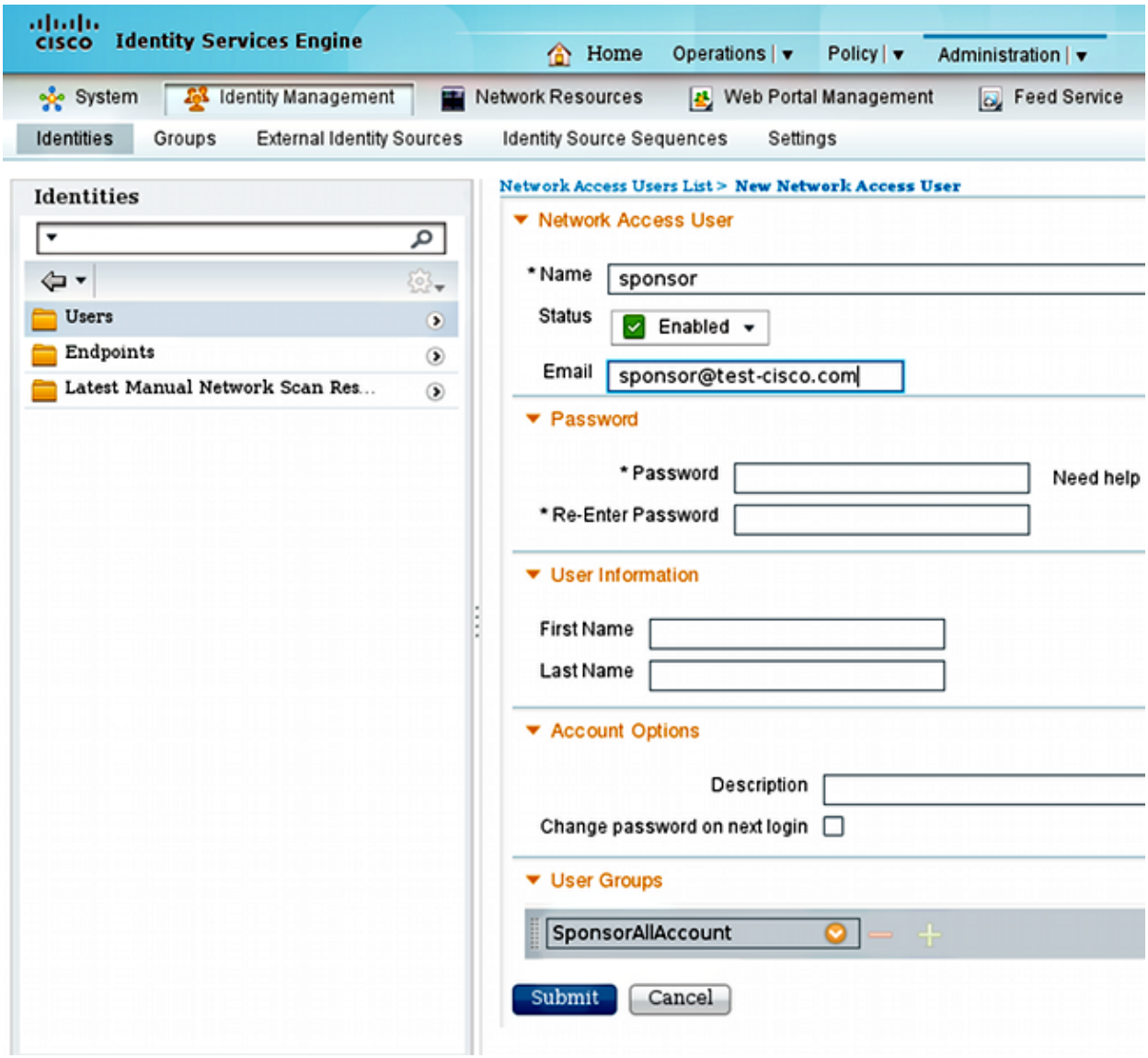
Each module of this solution (Postfix, Kannel smsbox, and Kannel bearerbox) can be installed on a separate server. For simplicity in this example, it is configured on the same server.

## Configurations

### ISE

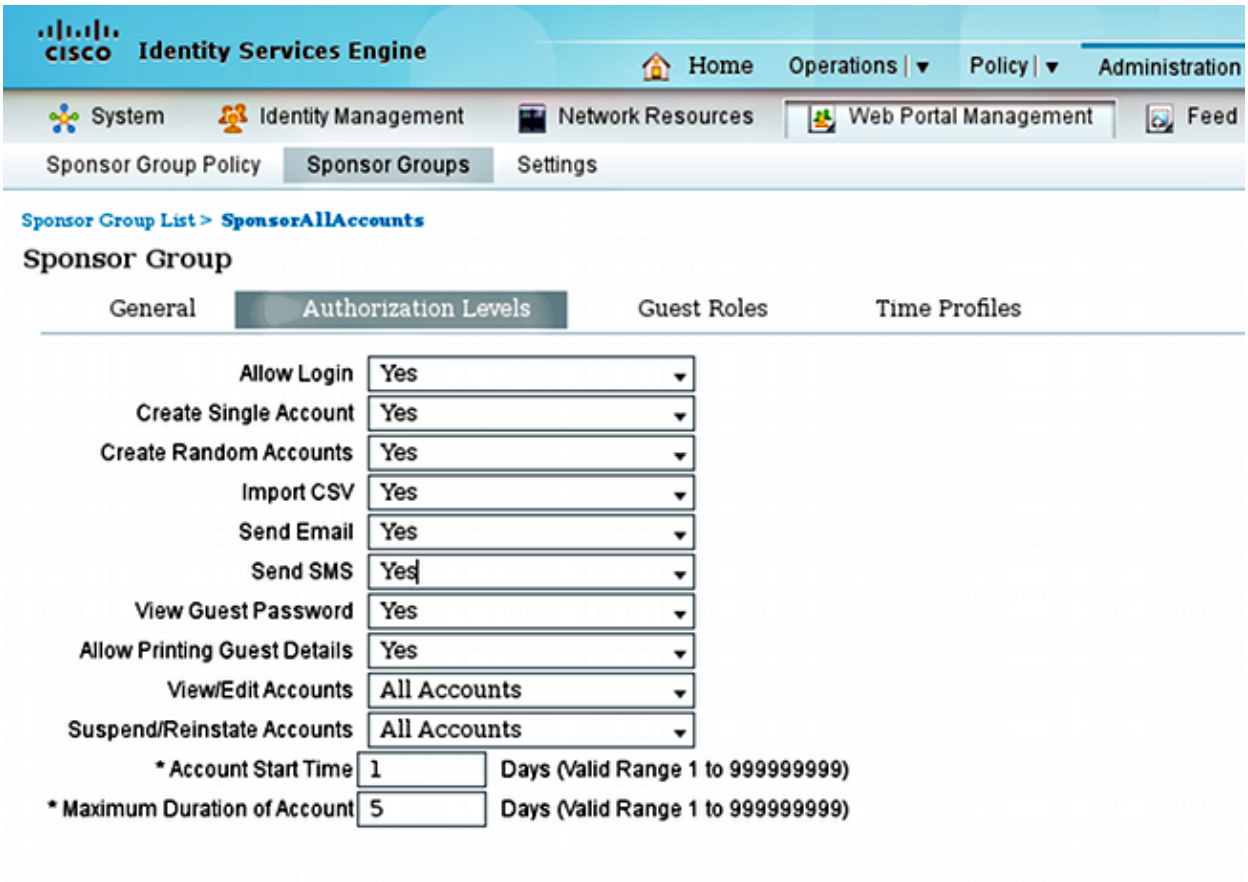
Complete these steps in order to configure the ISE.

1. Configure the sponsor portal user. In this example, the default ISE configuration is used, and the user is placed in the ***SponsorAllAccount*** group:



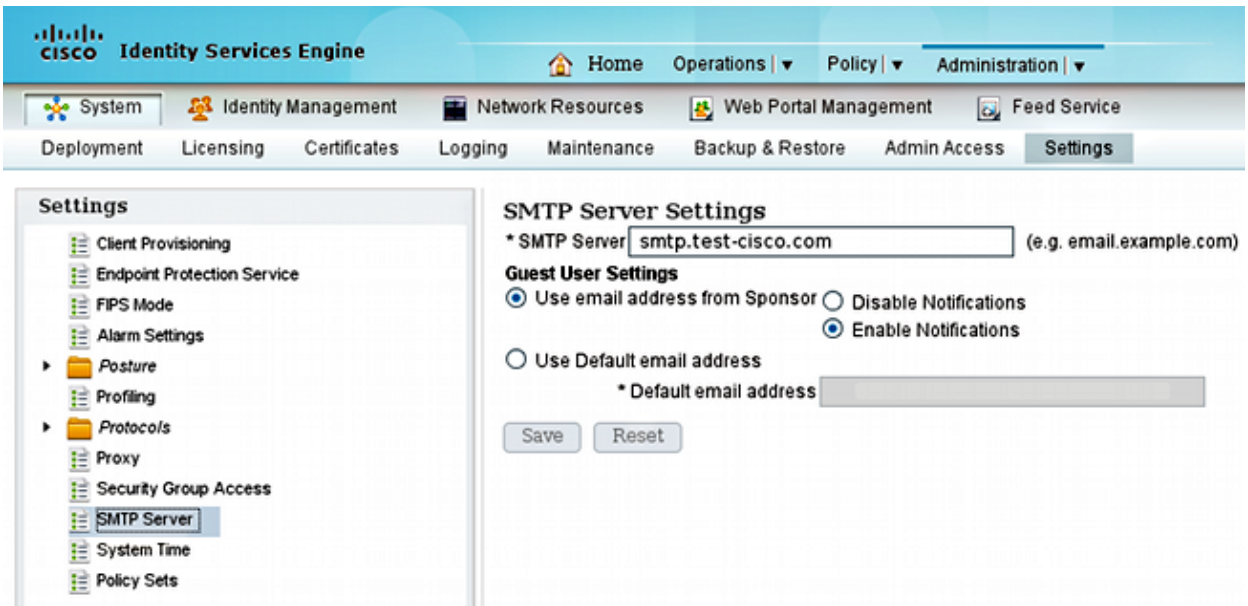

The email for the sponsor user can be configured later from the Sponsor Portal.

2. In order to be able to send SMS notifications, edit the default privileges for the ***SponsorAllAccount*** group:




By default, the ***Send SMS*** privilege is disabled.

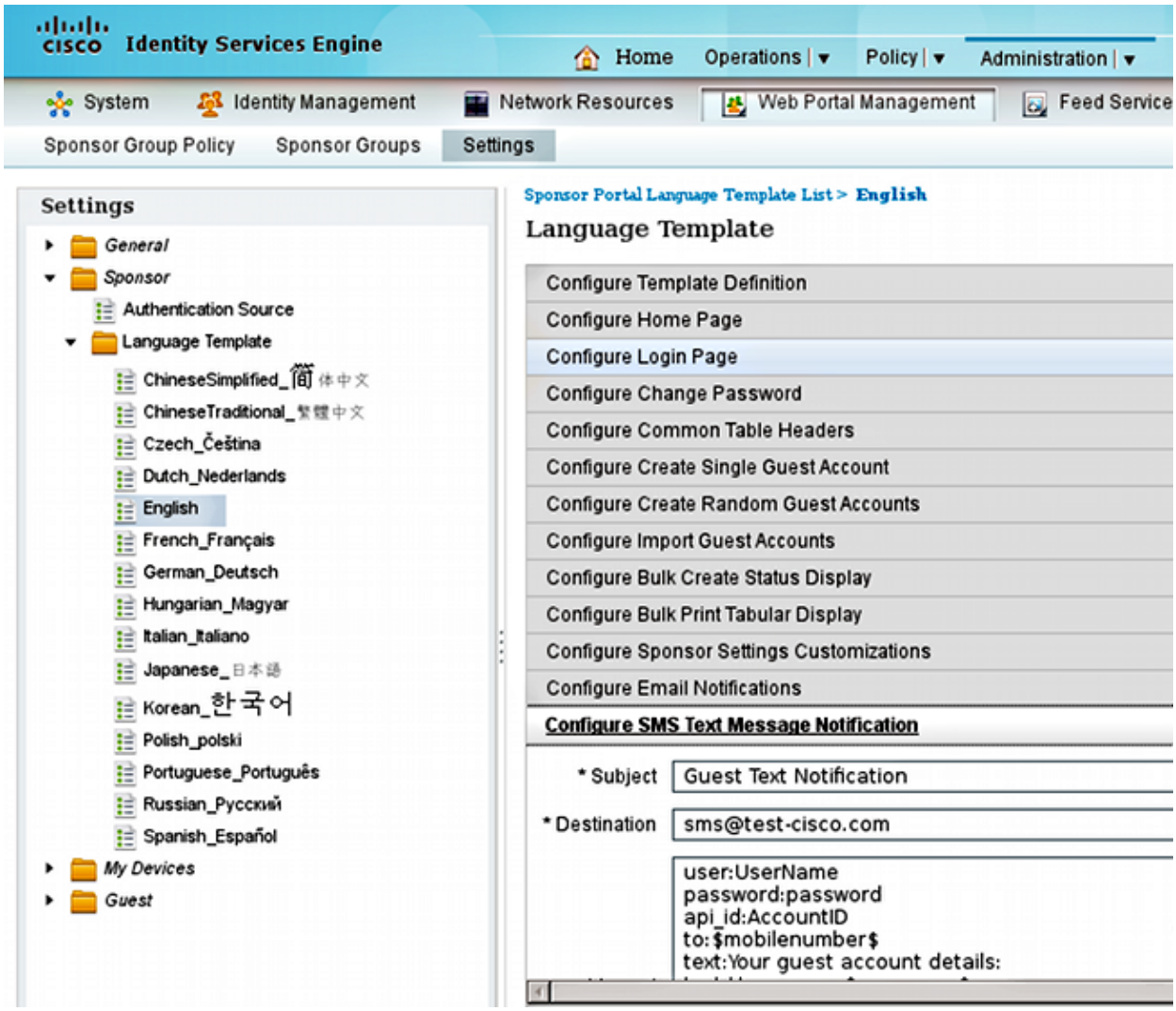
3. Configure the SMTP server, and make sure that the DNS settings are correct.




All notification emails are sent to the ***smtp.test−cisco.com*** host. ISE does not try to check the DNS

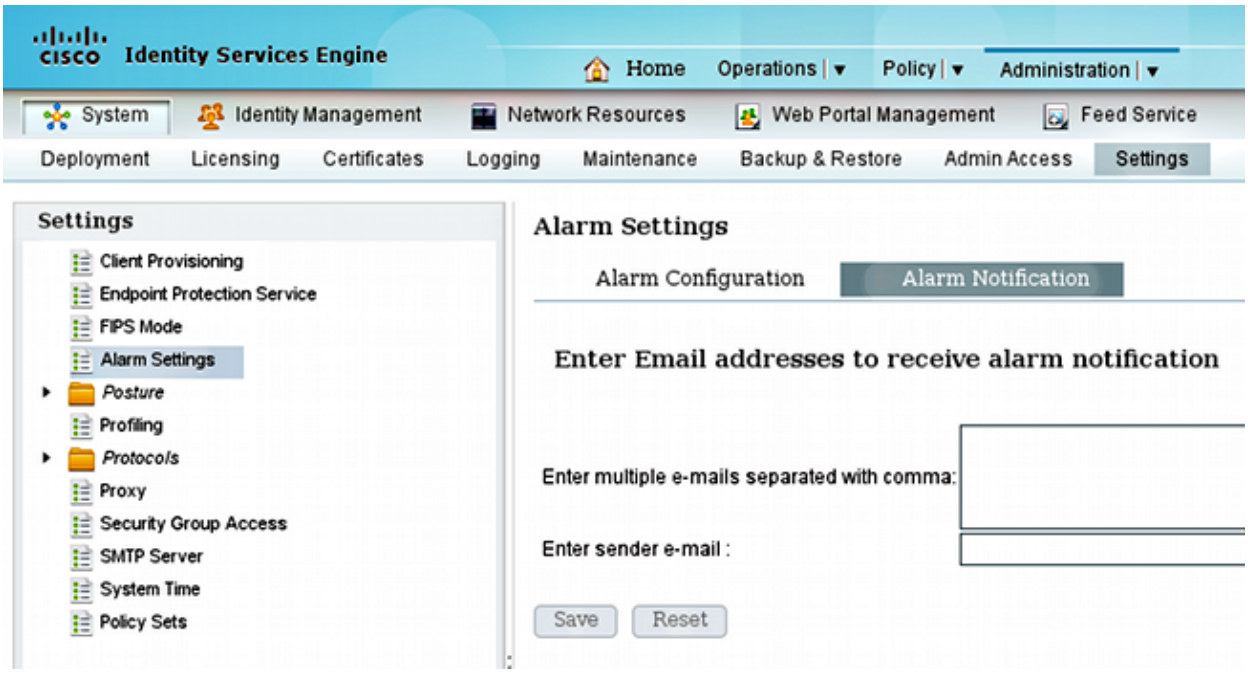
MX records for configured domains (this SMTP server is treated as a relay).

4. Customize the email that is sent as the SMS notification.




5. Configure the destination email address, which is the only setting that is not left as default. All of the notifications are sent via an SMTP server configured earlier with the ***To*** field set as ***sms@test−cisco.com***.

***Note***: It is possible to configure ISE in order to send alert notifications via the email. This can also be integrated with the proposed solution in order to send the alerts as SMS to mobile phones. Cisco advises that you use a separate account on the Postfix server for this (for example, ***alert@test−cisco.com***).




### Postfix

Postfix is an SMTP server that receives emails from ISE. The default configuration is used except for a few minor changes. Complete these steps in order to configure it.

1. Configure Postfix in order to be the local destination for the ***test−cisco.com*** domain. It is important to also configure a local delivery agent: maildrop. Here are the necessary changes in the main.cf:

```
myhostname = smtp.test−cisco.com
mydomain = test−cisco.com
mydestination = $myhostname, $mydomain, localhost
local_transport = maildrop
```

2. The next step is to activate maildrop in the master.cf. Change the correct line in the master.cf:

```
maildrop  unix  −       n       n       −       −       pipe
  flags=DRhu user=vmail argv=/usr/bin/maildrop −d ${user}
```

   Because it is a simple deployment without virtual domains, the ***{user}*** parameter is used instead of the default ***{recipient}*** parameter.

3. Configure the local account SMS that is used in order to receive the emails:

```
neptun ~ # useradd sms
neptun ~ # passwd sms
New password:
BAD PASSWORD: it is too simplistic/systematic
Retype new password:
passwd: password updated successfully
neptun ~ # chown −R sms:sms /home/sms/
```

Right now, all of the emails should be correctly delivered to the SMS user. The maildir structure is created automatically when it first receives email.

### Maildrop with Mailfilter

Just before the delivery, maildrop searches for ***.mailfilter*** in the home directory for the user. If that file is found, the script is executed. The privileges for the file should be limited to user only:

```
neptun sms # touch /home/sms/.mailfilter
neptun sms # chmod go-rwx /home/sms/.mailfilter
```

Here is the content of the file:

```
# Mailfilter script for parsing ISE SMS messages
# Author: Michal Garcarz at cisco.com
# Date: 1 Dec 2013

#DEFAULT="$HOME/.maildir/"
DATE=`date`
SHELL="/bin/bash"

# Our log file
logfile "/home/sms/maildrop.log"

# Our verbosity in the log file
VERBOSE="5"

log "-------------SMS MAILFILTER LOG-----------"
log "Email received at: $DATE"

if (/^Subject:.*Guest.*Text.*Notification.*/)
{
        log "Email processed by script sending SMS via Kannel"

        USERNAME=""
        PASSWORD=""
        TO=""
        if (/^text:Username:(.*)/:b)
        {
            log "Username exists $MATCH1"
            USERNAME=$MATCH1
        }
        if (/^text:Password:(.*)/:b)
        {
            log "Password exists $MATCH1"
            PASSWORD=$MATCH1
        }
        if (/^to:(.*)/:b)
        {
            log "Mobile phone exists $MATCH1"
            TO=$MATCH1
        }

        if ($USERNAME ne "" && $PASSWORD ne "" && $TO ne "")
        {
            log "Sending via HTTP to kannel username=$USERNAME password=$PASSWORD to=$TO"
            DATA="ISE Guest portal Username: $USERNAME Password: $PASSWORD"

            #also curl can be used instead of wget
            xfilter "wget -O/dev/null \"http://192.168.112.100:13013/cgi-bin/sendsms?username=
```

```
            tester&password=foobar&to=$TO&text=$DATA\" >> /tmp/maildrop-kannel.log 2>>
            /tmp/maildrop-kannel.log"
        }

        #deliver to maildir (not used since xfilter returns !=0)
        to $DEFAULT/
}
```

The script:

- Checks if the subject is the same as what is configured on the ISE
- Reads the username and password to fields for email body (the default template from the ISE is used)
- Calls an external program if all of the fields exist: ***wget*** in order to send HTTP GET to Kannel smsbox with all of the parameters. Notice that specific credentials are used in the URL (username=tester&password=foobar). These are the credentials of the user configured in Kannel with the privileges to send SMS.

There are two log files here:

- ***/home/sms/maildrop.log*** − logs from execution of the script
- ***/tmp/maildrop−kannel.log*** − logs from execution of wget

### Kannel

Both smsbox and bearerbox can be configured from the single file. This configuration uses the external SMPP server for delivery. It is easy to find multiple services on the web if you search for the ***smpp sms service provider*** phrase. The configuration is simple, because there is no need to receive and route SMS messages. This solution is only for sending and uses one SMPP provider.

Here is an excerpt from the ***/etc/kannel/kannel.conf***:

```
#bearerbox
group = core
admin-port = 13000
admin-password = bar
smsbox-port = 13001
log-level = 0
log-file = "/var/log/kannel/kannel.log"
access-log = "/var/log/kannel/access.log"

# SMSC SMPP
group = smsc
smsc = smpp
host = ****.com
port = 1775
smsc-username = 4ljt7wi3
smsc-password = ******
system-type =
address-range =

# SMSBOX SETUP
group = smsbox
bearerbox-host = localhost
sendsms-port = 13013
sendsms-chars = "0123456789 +-"
global-sender = 12345
log-file = "/var/log/kannel/smsbox.log"
log-level = 0
access-log = "/var/log/kannel/access.log"
```

```
white-list =
black-list =
reply-couldnotfetch =
reply-couldnotrepresent =
reply-requestfailed =
reply-emptymessage =

# SEND-SMS USERS, this credentials has been used in wget script
group = sendsms-user
username = tester
password = foobar
user-deny-ip = "*.*.*.*"
user-allow-ip = "192.168.*.*"

# SMS SERVICE Default
# there should be default always (this is for receiving SMS messages - not used)
group = sms-service
keyword = default
text = "No service specified"
```

It is possible to attach a mobile phone via USB and configure GSM SMSC:

```
group = smsc
smsc = at                  #type = GSM
smsc-id = usb0-modem
my-number = 1234
modemtype = auto           #types: wavecom, siemens, siemens-tc35, falcom,
nokiaphone, ericsson
device = /dev/ttyUSB0      #phone device seen on server
```

On most phones, it is also required to activate modem functionality; for example, in Android Version 2.2 and later, it is enabled in Settings/Tethering and Portable Hotspot/USB tethering.

Remember to run both bearerbox and smsbox. Here is an example:

```
neptun ~ # /etc/init.d/kannel-bearerbox start
 * Starting kannel bearerbox ...                                        [ ok ]
neptun ~ # /etc/init.d/kannel-smsbox start
 * Starting kannel smsbox ...                                           [ ok ]

neptun ~ # netstat -atcpn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address     Foreign Address    State    PID/Program name
tcp        0      0 0.0.0.0:13013     0.0.0.0:*          LISTEN   24170/smsbox
tcp        0      0 0.0.0.0:13000     0.0.0.0:*          LISTEN   24151/bearerbox
tcp        0      0 0.0.0.0:13001     0.0.0.0:*          LISTEN   24151/bearerbox
```

Bearerbox must have at least one SMSC configured in order to start.

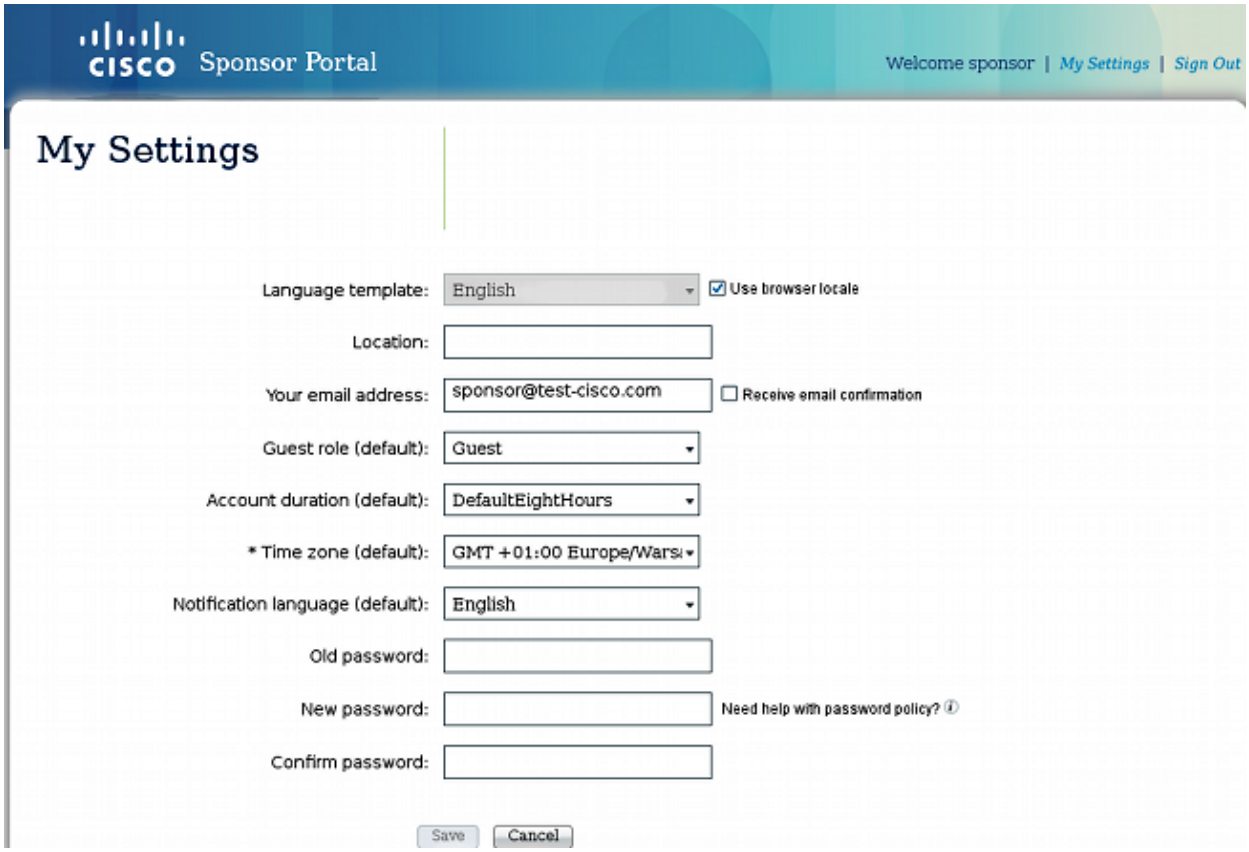
# Verify

Use this section in order to confirm that your configuration works properly.

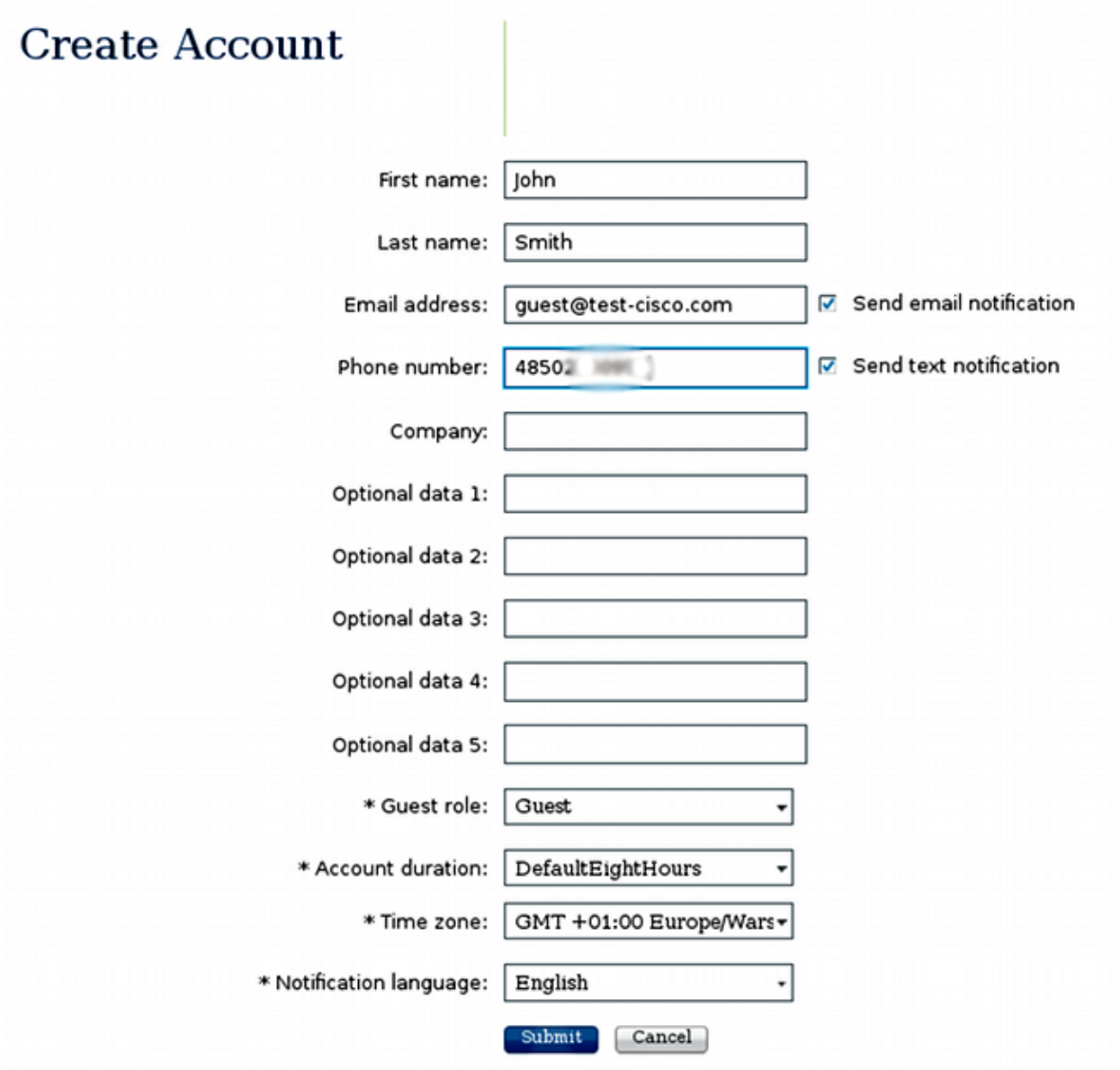
## ISE

The default port for the Sponsor Portal is used (***8443***). The sponsor logs into ***https://ise.test-cisco.com:8443/sponsorportal/***.

Make sure that the sponsor has an email address assigned in ***My Settings***:




Create the guest account with an SMS notification:




You recieve confirmation that the guest account was successfully created:


ISE should send an email to the configured SMTP server.

## Postfix

The SMTP server receives the message and uses maildrop in order to deliver it to the local account (***sms@test-cisco.com***). Here is an excerpt from the ***/var/log/messages***:

```
Nov 30 22:39:47 neptun postfix/smtpd[18460]: connect from unknown[192.168.112.1]
Nov 30 22:39:47 neptun postfix/smtpd[18460]: 2B36030B32: client=unknown
[192.168.112.1]
Nov 30 22:39:47 neptun postfix/cleanup[18463]: 2B36030B32: message-id=
<563762958.941385847586377.JavaMail.root@ise2>
Nov 30 22:39:47 neptun postfix/qmgr[32658]: 2B36030B32: from=<sponsor@test-cisco.com>,
size=689, nrcpt=1 (queue active)
Nov 30 22:39:47 neptun postfix/pipe[18464]: 2B36030B32: to=<sms@test-cisco.com>,
relay=maildrop, delay=0.18, delays=0.14/0/0/0.04, dsn=2.0.0, status=sent (delivered
via maildrop service)
```

## Maildrop

Before you send the email to the SMS, the maildir directory executes ***/home/sms/.mailfilter***, which performs a specific action.

Here is an excerpt from the /***home/sms/maildrop.log***:

```
------------SMS MAILFILTER LOG-----------
Email received at: Sat Nov 30 22:39:47 CET 2013
Email processed by script sending SMS via Kannel
Username exists  jsmith02
Password exists  t6ub79_6r
Mobile phone exists 4850xxxxxxx
Sending via HTTP to kannel username= jsmith02 password= t6ub79_6r to=4850xxxxxxx
```

## Mailfilter

The mailfilter script reads all of the data and executes ***xfilter***, which calls ***wget*** in order to pass all of the parameters to Kannel.

Here is an excerpt from the ***/tmp/maildrop−kannel.log***:

```
−−2013−11−30 22:39:47−−  http://192.168.112.100:13013/cgi−bin/sendsms?username=
tester&password=foobar&to=4850xxxxxxx&text=ISE%20Guest%20portal%20Username:
%20%20jsmith02%20Password:%20%20t6ub79_6r
Connecting to 192.168.112.100:13013... connected.
HTTP request sent, awaiting response... 202 Accepted
Length: 24 [text/html]
Saving to: `/dev/null'

     0K                                                       100% 1.14M=0s

2013−11−30 22:39:47 (1.14 MB/s) − `/dev/null' saved [24/24]
```

The HTTP GET request is accepted.

## Kannel

The Kannel smsbox reports that it received an HTTP request from ***wget*** and it sent that request to the bearerbox in order to deliver the SMS.

Here is an excerpt from the ***/var/log/kannel/smsbox.log***:

```
2013−11−30 22:39:47 [18184] [5] INFO: smsbox: Got HTTP request </cgi−bin/sendsms>
from <192.168.112.100>
2013−11−30 22:39:47 [18184] [5] INFO: sendsms used by <tester>
2013−11−30 22:39:47 [18184] [5] INFO: sendsms sender:<tester:12345>
(192.168.112.100) to:<4850xxxxxxx> msg:<ISE Guest portal Username:
jsmith02 Password:  t6ub79_6r>
2013−11−30 22:39:47 [18184] [5] DEBUG: Stored UUID fd508632−9408−49e1−9eda−
3ce8d4b939d4
2013−11−30 22:39:47 [18184] [5] DEBUG: message length 57, sending 1 messages
2013−11−30 22:39:47 [18184] [5] DEBUG: Status: 202 Answer: <Sent.>
2013−11−30 22:39:47 [18184] [5] DEBUG: Delayed reply − wait for bearerbox
2013−11−30 22:39:47 [18184] [0] DEBUG: Got ACK (0) of fd508632−9408−49e1−9eda−
3ce8d4b939d4
```

The Kannel bearerbox connects to the remote SMPP server and sends the message successfully.

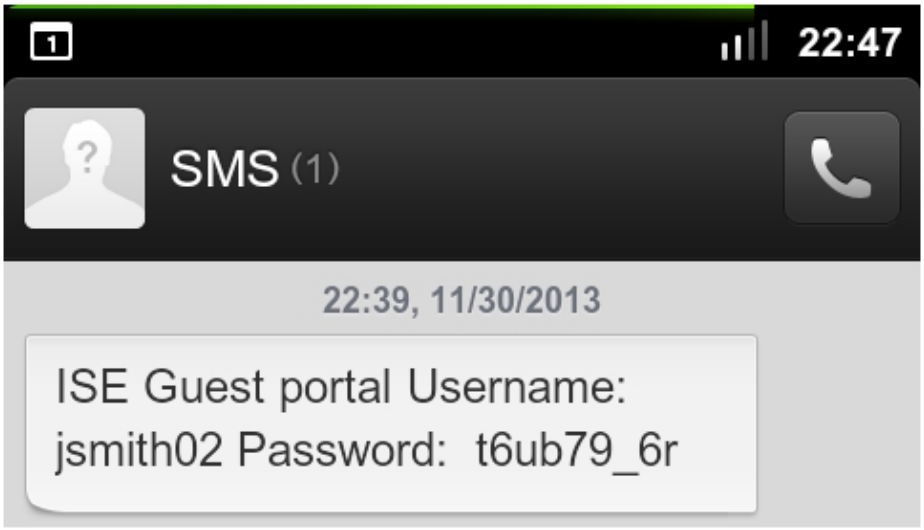
Here is an excerpt from the ***/var/log/kannel/kannel.log***:

```
2013-11-30 22:39:47 [18165] [8] DEBUG: boxc_receiver: sms received
2013-11-30 22:39:47 [18165] [8] DEBUG: send_msg: sending msg to box: <127.0.0.1>
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3:]:
throughput (0.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:*****.com:1775/0:4ljt7wi3:]:
Sending PDU:
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU 0x2056bf0 dump:
2013-11-30 22:39:47 [18165] [6] DEBUG:   type_name: submit_sm
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_id: 4 = 0x00000004
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_status: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sequence_number: 5 = 0x00000005
2013-11-30 22:39:47 [18165] [6] DEBUG:   service_type: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr_ton: 2 = 0x00000002
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr_npi: 1 = 0x00000001
2013-11-30 22:39:47 [18165] [6] DEBUG:   source_addr: "12345"
2013-11-30 22:39:47 [18165] [6] DEBUG:   dest_addr_ton: 2 = 0x00000002
2013-11-30 22:39:47 [18165] [6] DEBUG:   dest_addr_npi: 1 = 0x00000001
2013-11-30 22:39:47 [18165] [6] DEBUG:   destination_addr: "4850xxxxxxx"
2013-11-30 22:39:47 [18165] [6] DEBUG:   esm_class: 3 = 0x00000003
2013-11-30 22:39:47 [18165] [6] DEBUG:   protocol_id: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   priority_flag: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   schedule_delivery_time: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   validity_period: NULL
2013-11-30 22:39:47 [18165] [6] DEBUG:   registered_delivery: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   replace_if_present_flag: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   data_coding: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sm_default_msg_id: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sm_length: 57 = 0x00000039
2013-11-30 22:39:47 [18165] [6] DEBUG:   short_message:
2013-11-30 22:39:47 [18165] [6] DEBUG:    Octet string at 0x205ec70:
2013-11-30 22:39:47 [18165] [6] DEBUG:      len:  57
2013-11-30 22:39:47 [18165] [6] DEBUG:      size: 58
2013-11-30 22:39:47 [18165] [6] DEBUG:      immutable: 0
2013-11-30 22:39:47 [18165] [6] DEBUG:      data&colon; 49 53 45 20 47 75 65 73
74 20 70 6f 72 74 61 6c   ISE Guest portal
2013-11-30 22:39:47 [18165] [6] DEBUG:      data&colon; 20 55 73 65 72 6e 61 6d
65 3a 20 20 6a 73 6d 69    Username:  jsmi
2013-11-30 22:39:47 [18165] [6] DEBUG:      data&colon; 74 68 30 32 20 50 61 73
73 77 6f 72 64 3a 20 20   th02 Password:
2013-11-30 22:39:47 [18165] [6] DEBUG:      data&colon; 74 36 75 62 37 39 11 36
72                        t6ub79.6r
2013-11-30 22:39:47 [18165] [6] DEBUG:    Octet string dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:******.com:1775/0:4ljt7wi3:]:
throughput (1.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:******.com:1775/0:4ljt7wi3:]:
throughput (1.00,0.00)
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:******.com:1775/0:4ljt7wi3:]:
Got PDU:
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU 0x2056bf0 dump:
2013-11-30 22:39:47 [18165] [6] DEBUG:   type_name: submit_sm_resp
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_id: 2147483652 = 0x80000004
2013-11-30 22:39:47 [18165] [6] DEBUG:   command_status: 0 = 0x00000000
2013-11-30 22:39:47 [18165] [6] DEBUG:   sequence_number: 5 = 0x00000005
2013-11-30 22:39:47 [18165] [6] DEBUG:   message_id: "4128473611307259"
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP PDU dump ends.
2013-11-30 22:39:47 [18165] [6] DEBUG: SMPP[SMPP:******.com:1775/0:4ljt7wi3:]:
throughput (1.00,0.00)
```

Notice that the source address is set as ***12345***. This setting does not matter. The external SMPP server rewrites that value. It is possible to buy additional service in order to be presented differently.

### Guest Phone

The guest user receives an SMS:




## Troubleshoot

This section provides information you can use in order to troubleshoot your configuration.

### ISE

You might encounter this error when you create a guest account: Unable to send a text message to the following guest users: xxxx. You must add your email address to the settings page. If you receive that error message, verify the sponsor email address.

## Related Information

- ***Cisco Identity Services Engine − Users Guide 1.2***
- ***Kannel documentation***
- ***Postfix documentation***
- ***Technical Support & Documentation − Cisco Systems***

Updated: Dec 23, 2013 Document ID: 116918