

Forum:[Hack Forums](#)

Thread: [SS7 TOOL - A COMPREHENSIVE GUIDE FOR LINUX USERS](#)

Clumsy - November 13, 2025, 4:10 am

SS7 TOOL - A COMPREHENSIVE GUIDE FOR LINUX USERS

Hello everyone

Before diving into my brief tutorial on utilizing the SS7 Stack let's cover some fundamental details about SS7

Signaling System 7 (SS7)

Signaling System 7 (SS7) is a globally recognized network signaling protocol that enables independent (common channel) signaling for call setup billing routing and data exchange among nodes in the public switched telephone network (PSTN). SS7 protocols are tailored for telephone system control and are exclusively accessible to telephone network operators

Common channel signaling (CCS) separates the content of phone calls from the signaling data used to establish the call known as "out-of-band" signaling. This approach dedicates one channel in a multi-channel network to manage account for and control traffic across all network channels

An SS7 network comprises Service Switching Points (SSPs) Signaling Transfer Points (STPs) and Service Control Points (SCPs). The SSP collects analog signaling data from the local line (endpoint) and converts it into SS7 messages. These messages are relayed through the SS7 network via STPs which route the packet toward its destination. For specialized tasks such as redirecting a call to a forwarded number the STP forwards the message to an SCP a database that uses the incoming message to identify associated numbers or features for the call

Within the SS7 protocol an address (e.g. dialed digits) lacks explicit routing information. This requires the Signaling Connection Control Part (SCCP) translation function which uses routing tables to convert an address (typically a phone number) into the actual destination address or the address of an SCP containing the necessary customer data

The use of common channel signaling allows intelligence to be distributed across network databases and processing points enabling companies to provide advanced intelligent network (AIN) services through specialized service development tools

The diagram illustrates the SS7 control signaling system's structure. A customer's telephone connects to a local switch end office (EO). The SSP integrated into the EO translates dialed digits and signaling indicators (e.g. off-hook or answer) into SS7 messages. The SS7 network routes these control packets to their destination via STPs using separate interconnection lines. For advanced services SCP databases handle requests. The diagram highlights distinct signaling and voice connections with multiple redundant links between switches STPs and databases to ensure telephone network reliability. Links in the SS7 system such as Access links (A-links)

Bridge links (B-links) Cross links (C-links) Diagonal links (D-links) Extended links (E-links) and Fully associated links (F-links) serve specific functions and message structures

Today I'll guide you through a simple tutorial on using SS7 tools without developing a full program. I'm using the Telscale open-source SS7 stack compiled by Akib Sayyed provided as a Java .jar file eliminating the need to build or download additional tools

Requirements

Linux OS with SCTP support

JRE 1.7 (Java SE Runtime Environment) or higher

Download Links

SS7 Assessment Tool SafeSeven.Zip

JRE 8 (Java SE Runtime Environment) JRE8_Linux.tar.gz

Note JDK 7u6 and later include the JavaFX SDK (version 2.2 or later) integrated into the standard JDK directory. For JavaFX details visit

<http://docs.oracle.com/javase/8/javase-clienttechnologies.html>

Installing the 64-bit JRE on Linux

This process installs the Java Runtime Environment (JRE) for 64-bit Linux using an archive binary file (.tar.gz)

File jre-8uversion-linux-x64.tar.gz

Steps

Download the file. Accept the license agreement before downloading. The archive binary can be installed by anyone (not only root users) in any writable location. Only the root user can install the JDK into the system location

Change directory to the desired installation location then move the .tar.gz archive binary to the current directory

Unpack the tarball and install the JRE

```
% tar zxvf jre-8uversion-linux-x64.tar.gz
```

The Java Development Kit files are installed in a directory called jdk1.8.0_version in the current directory

Delete the .tar.gz file to save disk space

Extract the safeseven files to your home directory. Open WireShark (preinstalled in Kali Linux). Select the "SCTP" Protocol and start capturing data packets. Open a Terminal. Below are commands for using the Java .jar files

Before Running Client on Actual SS7 Network

Edit the client_config file (update details in the highlighted section)

```
//Client
```

```
SERVER_IP="IP of STP you are connecting to"
```

```
CLIENT_IP="IP address provisioned for you in STP"
```

```
SERVER_PORT="STP port"
```

```
CLIENT_PORT="client provisioned port"
```

```
IS_SERVER=FALSE "should be always false"
```

```
Local_SPC="point code assigned to you"
```

```
Remote_SPC="point code of STP"
```

```
Local_SSN="local ssn"
```

```
Remote_SSN="remote ssn"
```

```
Routing_Context="routing context assigned to you by STP"
```

```
NETWORK_INDICATOR="Network indicator"
```

```
Local_GT="Local global title assigned to you"
```

```
Remote_GT="remote Global title you are testing"
```

Commands for Simulating SS7 Network

Simulating HLR

```
java -jar server.jar hlr_config
```

Simulating MSC/VLR

```
java -jar server.jar vlr_config
```

Running STP

```
java -jar STP.jar stp_config
```

Commands for Running SafeSeven

SMS Related Operations

```
java -jar SMS.jar client_config
```

USSD Related Operations

```
java -jar ussd.jar client_config
```

Call Related Operations

```
java -jar Call_Handling.jar client_config
```

Mobility Related Operations

```
java -jar Mobility.jar client_config
```

Steps to Intercept SMS Using MapSMS.jar

Step 1

Attacker sends a SendRoutingInfoForSM request addressing the MAP (Mobile Application Part) message by MSISDN (Target Phone Number)

HLR (Home Location Register) responds with its own address serving MSC address and IMSI (International Mobile Subscriber Identity a unique number defined in ITU-T Recommendation E.212 consisting of a Mobile Country Code (MCC) a Mobile Network Code (MNC) and a Mobile Station Identification Number (MSIN))

Step 2

Attacker registers the Target Phone Number on the fake MSC

HLR updates the location for the target number

HLR requests the real MSC to release memory

Step 3

Someone sends an SMS to the Target Number

MSC forwards the SMS to SMS-C

SMS-C queries HLR for the Target Number's location

HLR responds with the fake MSC address

SMS-C sends the SMS to the fake MSC and WireShark captures the SMS

That's it

For Further Development

Download these programs

Eclipse Javascript IDE
Xampp for Linux
JDK (Java Development Kit) 8

Finding Info for Accessing SS7 Network (Sending SRISM)

Resources <https://gitea.osmocom.org/ss7-in-c>