Description

In the current state, Mr.SIP comprises four sub-modules named as SIP-NES, SIP-ENUM, SIP-DAS and SIP-ASP. Since it provides a modular structure to developers, more modules will continue be added by the authors and it is open to be contributed by the open-source developer community.

SIP-NES needs to enter the IP range or IP subnet information. It sends SIP OPTIONS message to each IP addresses in the subnet and according to the responses outputs the potential SIP clients and servers on that subnet.

SIP-ENUM outputs which SIP users are valid according to the responses in that network by sending REGISTER messages to each client IP addresses on the output of SIP-NES.

SIP-DAS (DoS Attack Simulator) is a module developed to simulate SIP-based DoS attacks. It comprises four components: spoofed IP address generator, SIP message generator, message sender and scenario player. It needs outputs of SIP-NES (Network Scanner) and SIP-ENUM (Enumerator) along with some pre-defined files.

SIP-DAS basically generates legitimate SIP INVITE message and sends it to the target SIP component via TCP or UDP. It has three different options for spoofed IP address generation, i.e., manual, random and by selecting spoofed IP address from subnet. IP addresses could be specified manually or generated randomly. Furthermore, in order to bypass URPF filtering, which is used to block IP addresses that do not belong to the subnet from passing onto the Internet, we designed a spoofed IP address generation module. Spoofed IP generation module calculated the subnet used and randomly generated spoofed IP addresses that appeared to come from within the subnet.

In order to bypass automatic message generation detection (anomaly detection) systems, random “INVITE” messages are generated that contained no patterns within the messages. Each generated “INVITE” message is grammatically compatible with SIP RFCs and acceptable to all of the SIP components.

“INVITE” message production mechanism specifies the target user(s) in the “To” header of the message. This attack can be executed against a single user or against legitimate SIP users on the target SIP server as an intermediary step before the DoS attack. The legitimate SIP users are enumerated and written to a file. Next, they are placed randomly in the “To” header of the generated “INVITE” messages. “Via, “User-Agent, “From,” and “Contact” headers within an “INVITE” message were syntactically generated using randomly selected information from the valid user agent and IP address lists. The tag parameter in the “From” header, the branch and source-port parameters in the “Via” header, and the values in the “Call-ID” header are syntactically and randomly generated using the valid user agent list. In addition, the source IP addresses in the “Contact” and “Via” headers are also generated using IP spoofing.

UDP is used widely in SIP systems as a transport protocol, so attacks on the target server are implemented by sending the generated attack messages in the network using UDP. Also TCP can be used optionally. The message sender of SIP-DAS allows the optional selection of how many SIP messages could be sent during one second. The number of SIP messages sent in one second depended on the resources (CPU and RAM) of the attacker machine.

SIP-ASP (Attack Scenario Player) allows the development of various SIP-based DoS attack scenarios through the use of SIP-DAS as the framework.

License = GNU General Public License v3.0

Installation

Requirements

Supported Platforms:

*nix environment

Installation:
apt-get install figlet toilet python python-scapy
pip install netifaces ipaddress
git clone https://github.com/meliht/Mr.SIP.git

Usage

Parameters

Usage: mr.sip.py [--ns|--ds|--se] [OPTIONS]

General options:

-h,	--help	Show this help message and exit
--ns,	--network-scanner	Scan specified network for live connections.
--ds,	--dos-simulator	Spoof IP addresses manually.
--se,	--sip-enumerator	Spoof IP addresses manually.

NES Options:

--tn=TARGET_NETWORK,	--target-network=TARGET_NETWORK	Target network range to scan.
-i IP_LIST,	--ip-save-list=IP_LIST	Output file location to save live IP address. Default location is inside application folder ip_list.txt.

DAS Options:

--dm=DOS_METHOD,	--dos-method=DOS_METHOD	DoS packet type selection. options, invite, register, sip-invite, subscribe, cancel, bye or other custom method file name.
-c COUNTER,	--count=COUNTER	Counter for how many messages to send. If not specified, default is flood.
-l,	--lib	Use Socket library (no spoofing), default is Scapy
--di=DEST_IP,	--destination-ip=DEST_IP	Destination SIP server IP address.
--dp=DEST_PORT,	--destination-port=DEST_PORT	Destination SIP server port number. Default is 5060.
-r,	--random	Spoof IP addresses randomly.

-m,	--manual	Spoof IP addresses manually. If you choose manual IP usage, you have to specify a IP list via --manual-ip-list parameter.
-s,	--subnet	Spoof IP addresses from subnet.
--to=TO_USER,	--to-user=TO_USER	To User list file location. Default is toUser.txt.
--fu=FROM_USER,	--from-user=FROM_USER	From User list file location. Default is fromUser.txt.
--su=SP_USER,	--sp-user=SP_USER	SP User list file location. Default is spUser.txt.
--ua=USER_AGENT,	--user-agent=USER_AGENT	User Agent list file location. Default is userAgent.txt.
--il=MANUAL_IP_LIST,	--manual-ip-list=MANUAL_IP_LIST	IP list file location.

Usage:
SIP-DAS usage-1: sudo ./mr.sip.py --ds --dm <sip_method_name> -c <number_of_packets> --di <server_ip> --dp <server_port> -r --to <to_user_file> --fu <from_user_file> --ua <user_agent_file> --su <sp_user_file>
SIP-DAS usage-2: sudo ./mr.sip.py --ds --dm <sip_method_name> -c <number_of_packets> --di <server_ip> --dp <server_port> -s --to <to_user_file> --fu <from_user_file> --ua <user_agent_file> --su <sp_user_file>
SIP-DAS usage-3: sudo ./mr.sip.py --ds --dm <sip_method_name> -c <number_of_packets> --di <server_ip> --dp <server_port> -m --to <to_user_file> --fu <from_user_file> --ua <user_agent_file> --su <sp_user_file> -il <client_ip_list>

SIP-NES Usage: sudo ./mr.sip.py --ns --tn <network_range> -i <file_location>

Usages examples:
SIP NES
./mr.sip.py –ns –tn 172.16.215.128/26 –i network_scan.txt

SIP-DAS
./mr.sip.py –ds –dm=invite –c 2 –di=172.16.215.130 –dp=5060 –r –to=toUser.txt –fu=fromUser.txt –ua=userAgent.txt –l

Tips for getting SIP trace:
ngrep -W byline -d eth0 port 5060
ngrep -W byline -d eth0 port 5060 -O capture_file
ngrep -W byline -d eth0 INVITE
tcpdump -i eth0 -n -s 0 port 5060
tcpdump -i eth0 -n -s 0 port 5060 -vvv -w /home/capture_file_name
tcpdump -nqt -s 0 -A -i en0 port 5060

TODO

Developments to be made:

1. SIP-ENUM will be integrated
2. SIP-ASP will be integrated
    a. Incomplete INVITE transaction DDoS with non-responding destination attack
    b. Incomplete INVITE dialog DDoS without ACK attack
3. The features to add to SIP-DAS
    a. Fragmentation and custom MTU value set support
    b. Network level IP spoofing support
    c. Instrumentation
4. Verbose mode support
5. Input validation should be performed
6. The whole code should be reviewed for all possibilities and exceptions