

Forum:[Hack Forums](#)

Thread: [\[Tutorial\] ~ DNS Rebinding ~ Hacking Routers and Taking Names](#)

Goodies - December 18, 2013, 11:24 pm

Hey guys :D

Alright, this is going to be a tutorial about DNS rebinding, what it is, how to do it, and how to protect against it.

What you'll need: A Web Server

Register a NameServer on your domain (Free)

Rebind ([download here](#))

1. What is DNS Rebinding?

I'm so glad you asked. DNS rebinding occurs when a web server provides an IP that is different than its own, generally for potentially malicious reasons. The attack vector is providing the IP of the client in order to perform Cross Site Request Forgery attack. We will be using this to 'hack' routers. This works because somewhere between some and many consumer-grade routers are configured in such a way that, when you type the external IP address of the router from within the local LAN, it will take you to the router configuration page. The reason this works is because the router checks only where the packet is going, and not the interface through which it came in. This means that an internal host can access the router by the router's INTERNAL or EXTERNAL IP. However, an external device can not access the router at all (without DNS rebinding or some sort of attack).

Here's how it works:

Home User: 1.1.1.1

Web Server: 3.3.3.3 (example.com)

RealNameServer: 8.8.8.8

FakeNameServer: 3.3.3.3 (same as web server)

When a browser sends a request to a DNS server requesting the IP address of that server, the DNS server sends back all DNS records it has.

1.1.1.1 Me ---> RealNameServer ::: A Record for example.com

1.1.1.1 Me <--- RealNameServer ::: IPs For example.com: 3.3.3.3

Then the well-known TCP handshake:

1.1.1.1 Me --SYN--> WebServer 3.3.3.3

1.1.1.1 Me <--SYN/ACK-- WebServer 3.3.3.3

1.1.1.1 Me --ACK--> WebServer 3.3.3.3

1.1.1.1 Me <--HTML/JS-- WebServer 3.3.3.3

1.1.1.1 Me --FIN--> WebServer 3.3.3.3

1.1.1.1 Me <--FIN/ACK-- WebServer 3.3.3.3

1.1.1.1 Me --ACK--> WebServer 3.3.3.3

That's it! That's the absolute most basic form of a web request.

Okay, so DNS rebinding simply works like this:

Registered Nameserver: ns1.example.com [NS]

When we request an A record for the website, it provides two IP addresses.

www.example.com [A] ~~~~~> 3.3.3.3, 1.1.1.1

Set your nameserver to ns1.example.com in your registrar.

Now that you have that set up, we're ready to begin. When the client requests your website, it will be provided with two different IP addresses: that of your web server, and that of their router. When they connect to you, you provide them with malicious HTML/JavaScript to automatically create requests to log into their router and perform your desired configuration changes.

1.1.1.1 Me ---> FakeNameServer ::: A Record for example.com

1.1.1.1 Me <--- FakeNameServer ::: IPs for example.com : 3.3.3.3 and 1.1.1.1

As you can see, it adds whatever source address the request is coming from and says "use this as an IP"

1.1.1.1 Me --SYN--> WebServer 3.3.3.3

1.1.1.1 Me <--SYN/ACK-- WebServer 3.3.3.3

1.1.1.1 Me --ACK--> WebServer 3.3.3.3

1.1.1.1 Me <--HTML/JS-- WebServer 3.3.3.3 * This is MALICIOUS JAVASCRIPT

1.1.1.1 Me --FIN--> WebServer 3.3.3.3

1.1.1.1 Me <--RST-- WebServer 3.3.3.3

1.1.1.1 Me NEW IP ADDRESS FOR example.com: 1.1.1.1

1.1.1.1 Me --SYN--> Router 1.1.1.1

1.1.1.1 Me <--SYN/ACK-- Router 1.1.1.1

1.1.1.1 Me --ACK--> Router 1.1.1.1

1.1.1.1 Me <--HTML/JS-- Router 1.1.1.1 * This is the Router Login Page

1.1.1.1 Me --FIN--> Router 1.1.1.1

1.1.1.1 Me <--FIN/ACK-- Router 1.1.1.1

1.1.1.1 Me --ACK--> Router 1.1.1.1

3.3.3.3 Malicious Webserver >>> Proxy via the Malicious Javascript >>> Router 1.1.1.1

Now the owner of the malicious web server has access to the router log in page for as long as the browser is open. The key behind the effectiveness of DNS rebinding is that many routers have default passwords that users do not change (ie: admin/admin, admin/password, root/, admin/, etc...). Once you're in the router, enable remote administration (if accessible) and set a password (that you know) and you will be able to remote into the router. From there, you can run the same attack over and over, sniffing the network, DNS cache poisoning, or other network attacks.

2. Actually Running the Attack

Alright, as I said earlier, you must have the nameserver configured in your DNS manager and set your website's nameserver to itself.

Confused? Research DNS.

Note: If you are behind a NAT or Firewall, you must have ports 53, 80 and 81 "forwarded" and allowed.

The command:

```
./rebind -i eth0 -d example.com
```

-i = the interface through which incoming connections travel

-d example.com = the domain you have registered

[+] Starting DNS server on port 53

[+] Starting attack Web server on port 80

[+] Starting callback Web server on port 81

[+] Starting proxy server on 3.3.3.3:664

Now what you have to do is open your web browser and set the PROXY SETTINGS to your web server to whatever rebind tells you..

Proxy Settings:

Type: HTTP Proxy

IP: 3.3.3.3 (example.com server)

Port: 664

Now go into your browser and type:

<http://rebind/>

You will be redirected to the Rebind web interface. Once you've done that, just wait for someone to browse to example.com and hope their router is vulnerable xD Click the box that pops up and you're in! I recommend that once you get into the router, you enable some form of remote access.

3. Blocking This Attack

~ Router Config:

This attack has been mainly stopped or prevented by more secure router configurations. What allows this to work is the fact that services (like the HTTP server on your router) are bound to all interfaces and therefore are accessible by all IP's that it has. It will drop anything coming in the external port, but that doesn't matter because it is accessing the external IP from the internal LAN.

~ DNS Pinning:

DNS Pinning basically takes the first DNS response of a site and keeps it. It doesn't allow for updates... This is depreciated because DNS load balancing is vitally important to major sites, and it interferes with that. However, some sort of verification may be implemented to ensure that each site within a multi-answer DNS response is apart of that domain.

~ Simple Security Measure

Change your fucking password

Prioritize HTTPS. Using certificates will allow you to find out whether a secure connection is actually in existence.

nuvexy - December 18, 2013, 11:27 pm

wow much information, such hack, very word.

Very nice tutorial, though. Very well written!

Good job!

Kirian - December 18, 2013, 11:28 pm

Good tutorial, Goodies. You're such hq wow

Shinigami-san - December 18, 2013, 11:28 pm

wow much information, such hack, very word.

Goodies - December 18, 2013, 11:34 pm

You Dogefags make me sick...hail doge

Kirian - December 18, 2013, 11:35 pm

Your tutorials make me sick..

Spoiler ([Click to View](#)) Jokes, but take back what you said

Ginster - December 18, 2013, 11:38 pm

This is one of the coolest tutorials I've read on here so far.

Goodies - December 19, 2013, 1:38 am

(Dec 19, 2013 12:38 AM)Ginster Wrote: ([CITE](#)) This is one of the coolest tutorials I've read on here so far.

Aww <3 thanks.

I try hard to make informative tutorials :)

Private666 - December 19, 2013, 3:29 am

Fantastic tutorial once again HQ user

Goodies - December 26, 2013, 2:51 am

(Dec 19, 2013 04:29 AM)Syncing Wrote: ([CITE](#)) Fantastic tutorial once again HQ user

Hahaha thanks man :) Glad you enjoyed.

PoLoW - December 26, 2013, 3:02 am

This is such a HQ thread and HQ member , is it public or it's private ?

thanks for the informative thread

LiamsCPU - February 24, 2014, 10:23 am

Great tutorial, very HQ thread, thanks!

D2IZn-C - March 1, 2014, 4:11 pm

Great tutorial, I enjoyed it, and it was helpful... thank you

Mopex - March 1, 2014, 6:35 pm

Really good TuT, ive learnt alot from you since ive joined this forum.

Not to sound like a dick-Rider, but I look up to you;)

Goodies - March 1, 2014, 6:44 pm

(Mar 1, 2014 07:35 PM)Renegades Wrote: [\[CITE\]](#) Really good TuT, ive learnt alot from you since ive joined this forum.

Not to sound like a dick-Rider, but I look up to you;)

Oh man I really appreciate that :P Thank you!

mehere - August 21, 2014, 2:40 am

Goodies, another great tut, not just the how but the why and how to stop it.

Blackbullz - August 21, 2014, 3:02 am

Awesome tutorial ! Great job buddy. ?

bladedemon - November 12, 2014, 1:53 pm

This is quite informative, and it can even be used as another attack vector against, say, clients of a webpage you compromised, sort of post exploitation exploit ?

Hmmm, quite a bit of food for thought. Still, hat's off to you, sir ?

Bibl - December 20, 2014, 11:34 am

Great thread thanks brah Its really goood

P1R4T3 - December 21, 2014, 6:35 am

Awesome tutorial, but can you give me some more information?

What is the requirements for web server? OS and spec please :)

HiddenEyE - December 8, 2015, 4:41 am

i think i found this very useful. would try it out

LordDragon - July 17, 2020, 2:19 am

High quality tutorial,from one of HF best.