

Forum:[Hack Forums](#)

Thread: [\[Tutorial\] REVERSE SHELL SERIES - The Basics \(Part 1\) \[Tutorial\]](#)

TheCyb3rAlpha - April 26, 2020, 6:59 am

REVERSE SHELL SERIES - The Basics (Part 1)

Greetings to all HF members,

I'm introducing the reverse shell series for everyone who wants to learn about the reverse shell in depth. As me and my partner [??\\$???](#) have already written a book - [Hands On - Red Team Tactics](#) that has this chapter, we would like to expand the horizon for everyone to learn more about reverse shells with different case scenarios.

All the suggestions are welcome. Enjoy the tutorial!

In this tutorial, you'll learn about:

[+] Understanding the basics

[+] Shell Shoveling

[+] Introduction to Reverse Shells

Understanding the Basics

Spoiler ([Click to View](#))

A reverse connection (a.k.a egress connection, a.k.a backward connection) originates from the server placed inside a network (mostly internal network) and proceeds through the routers and switches (network devices) to a destination outside of the network. When the ingress traffic, i.e. the network traffic flowing towards the server (source -> destination) is filtered and blocked by the firewalls in place, a reverse connection can help bypass the firewall restrictions so that the server starts a connection to the client (destination -> source), i.e. generating egress traffic. A firewall usually blocks incoming connections on open ports but does not block outgoing traffic (Of course there are situations when the firewall also filters the egress traffic. In such a case a different approach is required which i'll cover later in this series).

A bind connection (a.k.a ingress connection, a.k.a forward connection) originates from the client to the server, a normal connection when made to connect to a destination IP. Generally, most of the servers have firewalls (hardware firewalls and software firewalls) in place to restrict the connection on the server. For example, a web server running on port 80/tcp will most probably have a firewall in place to restrict access to other ports open on the same server.

In a typical hacking scenario, a bind connection is when the attacker machine connects to the target server as shown in the following screenshot:

A reverse connection is when the target server initiates a connection to the attacker machine as shown in the following screenshot:

Shell shoveling

Spoiler ([Click to View](#))

It is a process where an attacker gains interactive shell access to a target. Generally, the interactive shell only works when the shell is opened locally on the system but in case an attacker has Remote Code Execution (RCE) vulnerability on the target server, he/she can get an interactive shell of the

target server using Shell Shoveling. This allows an attacker to bypass firewall rules on a target, as the target sends a request to the attacker's machine, and presents the shell there.

In a nutshell, a shell shovel is an act of redirecting the I/O of the shell (e.g. cmd.exe or sh shell) to a remote machine (in this case, an attacker).

Introduction to Reverse Shells

Spoiler ([Click to View](#))

A reverse shell is a type of shell in which the target server connects back to the attacker machine. For example, an attacker finds a target server with port 21/tcp, 80/tcp, and 443/tcp in OPEN state and the FTP service running on port 21/tcp is vulnerable. Let's say an attacker exploits this port in order to open another port 1337/tcp on the target server for shell connection, as shown in the following diagram:

The problem arises when the attacker tries to connect to the target server on port 1337/tcp. The attacker is not able to connect to port 1337/tcp. Why? Because the port 1337/tcp is not allowed in the firewall restrictions. The firewall can only allow port 21/tcp, 80/tcp and 443/tcp for incoming connections and it will block all other ports, as shown in the following diagram:

So in such a scenario, what do you think an attacker can do? Instead of opening port 1337/tcp on the target server, the attacker will open a port (let's say 1337/tcp) locally and execute a command on the target server which will connect back to the attacker machine. Hence, giving interactive shell access to the attacker.

Hope everyone likes it.

Checkout the second part here: [REVERSE SHELL SERIES - The Basics \(Part 2\)](#)

Stay connected and stay safe! Cheers ?

References

https://en.wikipedia.org/wiki/Reverse_connection

https://subscription.packtpub.com/book/networking_and_servers/9781788995238/5/ch05lv1sec42/introduction-to-reverse-shell-connections

<http://www.madirish.net/207>

??¢??? - April 26, 2020, 7:34 am

We hope everyone learns something out of it. More will be added to this soon

ceR0 - July 22, 2020, 7:59 am

Cheers mate keep it up! would love to see more of these tuts

SeQueL - July 29, 2020, 1:11 pm

Thx, first is very intresting so time for second part. It's sad, in tutorial section there is much spam and value threads like that is dissapear. But thanks a lot and wait for next tutorials!

SIM-5382664 - July 2, 2023, 9:44 am

thanks man apreciate it

SIM-5614029 - *February 18, 2025, 4:28 am*

Great first steps

Thank you

RoundLettuce - *February 27, 2025, 4:13 am*

Thank you bro. This is an excellent tutorial!