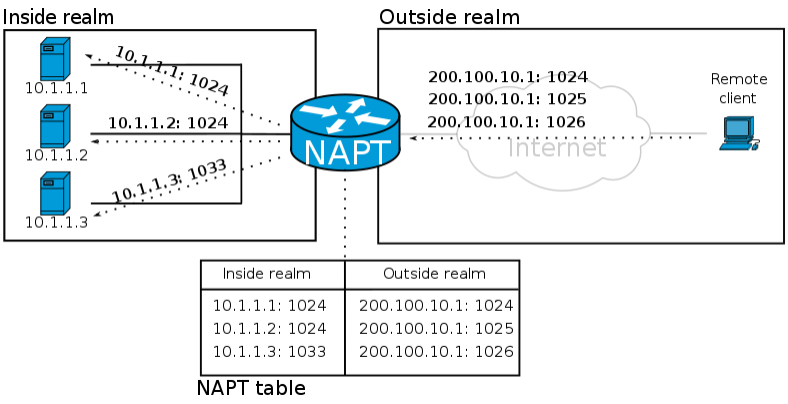
Port forwarding

In computer networking, **port forwarding** or **port mapping** is an application of network address translation (NAT) that redirects a communication request from one address and port number combination to another while the packets are traversing a network gateway, such as a router or firewall. This technique is most commonly used to make services on a host residing on a protected or masqueraded (internal) network available to hosts on the opposite side of the gateway (external network), by remapping the destination IP address and port number of the communication to an internal host.[1][2]



Port forwarding via NAT router

Purpose

Port forwarding allows remote computers (for example, computers on the Internet) to connect to a specific computer or service within a private local-area network (LAN).[3]

In a typical residential network, nodes obtain Internet access through a DSL or cable modem connected to a router or network address translator (NAT/NAPT). Hosts on the private network are connected to an Ethernet switch or communicate via a wireless LAN. The NAT device's external interface is configured with a public IP address. The computers behind the router, on the other hand, are invisible to hosts on the Internet as they each communicate only with a private IP address.

When configuring port forwarding, the network administrator sets aside one port number on the gateway for the exclusive use of communicating with a service in the private network, located on a specific host. External hosts must know this port number and the address of the gateway to communicate with the network-internal service. Often, the port numbers of well-known Internet services, such as port number 80 for web services (HTTP), are used in port forwarding, so that common Internet services may be implemented on hosts within private networks.

Typical applications include the following:

- Running a public HTTP server within a private LAN
- Permitting Secure Shell access to a host on the private LAN from the Internet
- Permitting FTP access to a host on a private LAN from the Internet
- Running a publicly available game server within a private LAN

Administrators configure port forwarding in the gateway's operating system. In Linux kernels, this is achieved by packet filter rules in the iptables or netfilter kernel components. BSD and macOS operating systems prior to Yosemite (OS 10.10.X) implement it in the Ipfirewall (ipfw) module while macOS operating systems beginning with Yosemite implement it in the Packet Filter (pf) module.

When used on gateway devices, a port forward may be implemented with a single rule to translate the destination address and port. (On Linux kernels, this is DNAT rule). The source address and port are, in this case, left unchanged. When used on machines that are not the default gateway of the network, the source address must be changed to be the address of the translating machine, or packets will bypass the translator and the connection will fail.

When a port forward is implemented by a proxy process (such as on application layer firewalls, SOCKS based firewalls, or via TCP circuit proxies), then no packets are actually translated, only data is proxied. This usually results in the source address (and port number) being changed to that of the proxy machine.

Usually only one of the private hosts can use a specific forwarded port at one time, but configuration is sometimes possible to differentiate access by the originating host's source address.

Unix-like operating systems sometimes use port forwarding where port numbers smaller than 1024 can only be created by software running as the root user. Running with superuser privileges (in order to bind the port) may be a security risk to the host, therefore port forwarding is used to redirect a low-numbered port to another high-numbered port, so that application software may execute as a common operating system user with reduced privileges.

The Universal Plug and Play protocol (UPnP) provides a feature to automatically install instances of port forwarding in residential Internet gateways. UPnP defines the Internet Gateway Device Protocol (IGD) which is a network service by which an Internet gateway advertises its presence on a private network via the Simple Service Discovery Protocol (SSDP). An application that provides an Internet-based service may discover such gateways and use the UPnP IGD protocol to reserve a port number on the gateway and cause the gateway to forward packets to its listening socket.

Types of port forwarding

Port forwarding can be divided into the following specific types: local, remote, and dynamic port forwarding.[4]

Local port forwarding

Local port forwarding is the most common type of port forwarding. It is used to let a user connect from the local computer to another server, i.e. forward data securely from another client application running on the same computer as a Secure Shell (SSH) client. By using local port forwarding, firewalls that block certain web pages are able to be bypassed.[5]

Connections from an SSH client are forwarded, via an SSH server, to the intended destination server. The SSH server is configured to redirect data from a specified port (which is local to the

host that runs the SSH client) through a secure tunnel to some specified destination host and port. The local port is on the same computer as the SSH client, and this port is the "forwarded port". On the same computer, any client that wants to connect to the same destination host and port can be configured to connect to the forwarded port (rather than directly to the destination host and port). After this connection is established, the SSH client listens on the forwarded port and directs all data sent by applications to that port, through a secure tunnel to the SSH server. The server decrypts the data, and then redirects it to the destination host and port.[6]

On the command line, “-L” specifies local port forwarding. The destination server, and two port numbers need to be included. Port numbers less than 1024 or greater than 49150 are reserved for the system. Some programs will only work with specific source ports, but for the most part any source port number can be used.

Some uses of local port forwarding:

- Using local port forwarding to Receive Mail [7]
- Connect from a laptop to a website using an SSH tunnel.

Remote port forwarding

This form of port forwarding enables applications on the server side of a Secure Shell (SSH) connection to access services residing on the SSH's client side.[8] In addition to SSH, there are proprietary tunnelling schemes that utilize remote port forwarding for the same general purpose.[9] In other words, remote port forwarding lets users connect from the server side of a tunnel, SSH or another, to a remote network service located at the tunnel's client side.

To use remote port forwarding, the address of the destination server (on the tunnel's client side) and two port numbers must be known. The port numbers chosen depend on which application is to be used.

Remote port forwarding allows other computers to access applications hosted on remote servers. Two examples:

- An employee of a company hosts an FTP server at their own home and wants to give access to the FTP service to employees using computers in the workplace. In order to do this, an employee can set up remote port forwarding through SSH on the company's internal computers by including their FTP server’s address and using the correct port numbers for FTP (standard FTP port is TCP/21) [10]

- Opening remote desktop sessions is a common use of remote port forwarding. Through SSH, this can be accomplished by opening the virtual network computing port (5900) and including the destination computer's address.[6]

Dynamic port forwarding

Dynamic port forwarding (DPF) is an on-demand method of traversing a firewall or NAT through the use of firewall pinholes. The goal is to enable clients to connect securely to a trusted server that acts as an intermediary for the purpose of sending/receiving data to one or many destination servers.[11]

DPF can be implemented by setting up a local application, such as SSH, as a SOCKS proxy server, which can be used to process data transmissions through the network or over the Internet. Programs, such as web browsers, must be configured individually to direct traffic through the proxy, which acts as a secure tunnel to another server. Once the proxy is no longer needed, the programs must be reconfigured to their original settings. Because of the manual requirements of DPF, it is not often used.[6]

Once the connection is established, DPF can be used to provide additional security for a user connected to an untrusted network. Since data must pass through the secure tunnel to another server before being forwarded to its original destination, the user is protected from packet sniffing that may occur on the LAN.[12]

DPF is a powerful tool with many uses; for example, a user connected to the Internet through a coffee shop, hotel, or otherwise minimally secure network may wish to use DPF as a way of protecting data. DPF can also be used to bypass firewalls that restrict access to outside websites, such as in corporate networks.

See also

- Firewall pinhole
- NAT traversal
- Packet forwarding
- Port address translation (PAT)
- Port triggering
- UDP Helper Address

- Secure Shell

References

1. *"Definition of: port forwarding" (https://www.pcmag.com/encyclopedia_term/0,1237,t=port+forwarding&i=49509,00.asp)* . *PC Magazine*. Retrieved 2008-10-11.
2. Rory Krause. *"Using ssh Port Forwarding to Print at Remote Locations" (http://www.linuxjournal.com/article/5462)* . *Linux Journal*. Retrieved 2008-10-11.
3. Jeff "Crash" Goldin. *"How to set up a home web server" (https://web.archive.org/web/20081004123716/http://www.redhat.com/magazine/022aug06/features/webserver/)* . *Red Hat*. Archived from *the original (http://www.redhat.com/magazine/022aug06/features/webserver/)* on 2008-10-04. Retrieved 2008-10-11.
4. *OpenSSH Port forwarding (https://help.ubuntu.com/community/SSH/OpenSSH/PortForwarding)*
5. *"Local and Remote Port Forwarding and the Reflection for Secure IT Client 7.1 or Higher - Tech Note 2433" (http://support.attachmate.com/techdocs/2433.html)* . Support.attachmate.com. 2012-11-09. Retrieved 2014-01-30.
6. *"SSH/OpenSSH/PortForwarding - Community Ubuntu Documentation" (https://help.ubuntu.com/community/SSH/OpenSSH/PortForwarding)* . Help.ubuntu.com. 2013-12-13. Retrieved 2014-01-30.
7. *"Example — Using Local Port Forwarding to Receive Mail (System Administration Guide: Security Services)" (http://docs.oracle.com/cd/E19683-01/806-4078/secsshuser-ex-42/index.html)* . Docs.oracle.com. Retrieved 2014-01-30.
8. *"Tunneling with Secure Shell - Appendix A: Remote Port Forwarding" (http://www.vandyke.com/solutions/port_forwarding/appendix_a.html)* . Vandyke.com. 2005-06-12. Retrieved 2014-01-30.
9. *"Local versus Remote Port Forwarding" (http://www.networkactiv.com/Pages/Local-versus-Remote-Port-Forwarding.html)* . NetworkActiv. Retrieved 8 June 2014.
10. *"FTP Port Number 21 - Port 21 TCP" (http://compnetworking.about.com/od/tcpip/p/port-numbers-21-ftp.htm)* . Compnetworking.about.com. 2013-12-19. Retrieved 2014-01-30.
11. *"DPF Mechanism" (http://pages.cs.wisc.edu/~sschang/firewall/dpf/mechanism.htm)* . Pages.cs.wisc.edu. Retrieved 2014-01-30.
12. *"SSH Dynamic Port Forwarding (Hacking Illustrated Series InfoSec Tutorial Videos)" (http://www.irongeek.com/i.php?page=videos/sshdynamicportforwarding)* . Irongeek.com. Retrieved 2014-01-30.

External links

- Alan Stafford. "Warp Speed Web Access: Sharing the Bandwidth" (https://web.archive.org/web/20080318010126/http://pcworld.about.com/magazine/1901p102id35287.htm) . PC World. Archived from the original (http://pcworld.about.com/magazine/1901p102id35287.htm) on 2008-03-18. Retrieved 2008-10-11.

- Using UPnP for Programmatic Port Forwardings and NAT Traversal (http://www.codeproject.com/KB/IP/PortForward.aspx) – Free software which uses UPnP and the Internet Gateway Device Protocol (IGD) to automate port forwarding
- TCP forwarding source code in C# (http://blog.brunogarcia.com/2012/10/simple-tcp-forwarder-in-c.html) – Source code in C# explaining/PoC TCP forwarding.
- Open.NAT (https://github.com/lontivero/Open.Nat) – Lightweight and easy-to-use .NET class library to allow port forwarding in NAT devices that support UPNP and PMP.
- Port Checker (https://www.portcheckers.com) Port Forwarding Testing Tool

---

Last edited 1 month ago by 122.162.146.4