**Host, Network and Virtual Firewalls**

**Most firewalls work on the network or transport layers to accept or reject traffic based on port, protocol and address**

* This means you can allow specific traffic out of your network only if you want
* TCP:HTTP:80
* TCP:HTTPS:443
* UDP:DNS:53

Can restrict ip addresses or ip ranges

**Deep Packet Inspection (DPI)**

* Specific protocol and port-based rules
* Will allow HTTPS traffic on 443, then inspect the traffic and make sure that is desirable and fits to the rules specified
* Confirms that connection through 443 is actually HTTPS and not HTTP or Tor or something, and then will let it through
* Firewalls that work on the network level (layer 4 and below) can’t do this, it just sees port 443 and allows the traffic through
* Only firewalls on level 5,6 and 7 (the application level) will perform DPI

Inbound/Ingress filtering

NATs stop inbound traffic from real internet IPs from interacting with private local IPs unless port forwarding or DMZ is set up

Devices running their own OS can have their own firewalls (Host based)

* Laptop
* Phone

NAT doesn’t protect you from untrusted devices on a shared network e.g., internet cafe

So, a host firewall is necessary to stop traffic from the local network, through to your device

All Firewalls should have an implicit deny all external inbound traffic unless specifically required

* For Home routers, this isn’t required because of NAT
* Is a required configuration for host-based firewalls

Outbound/Egress filtering

DNS leaks

VPN leaks

Malware communicating to its external command centre

Phone-home feature to Microsoft

Malware that executes code on your machine, will also need the code to communicate it back to the adversary

* Malware doesn’t use inbound connections to communicate because it knows that they are usually blocked by NAT and firewalls
* Reverse shell or reverse connection is necessary
* Attackers will communicate outbound instead, once they have gotten a foot in your machine
* Shell code – code designed specifically to communicate out and gain remote control
  + To create a reverse shell
* A reverse shell often creates a shell command

Pentestmonkey.co.uk reverse shell cheat sheet

Network isolation / segmentation

* Can be used to block traffic between devices
* Prevents propagation of an attack from an internal device

IoT devices are more likely to be backdoored so its important to isolate those untrusted devices from your trusted devices like your laptop

Network-based firewalls

Routes traffic through one network to another

* Usually 2 networks as a minimum
* Filters local traffic (LAN) to the internet (in a home network)

Could be set up on the router, using iptables

* Easiest way to get network firewall: install custom firmware with iptables

One disadvantage of a network firewall is that malware uses the ports and protocols that are allowed out communicate out like the internet protocol ports i.e., 443, 53, 80 etc.

* Therefore, outbound connections are not protected or blocked
* Can’t distinguish between good traffic and bad traffic

Firewalls that work on the application level can perform DPI, like PfSense or SmoothWall

* Can inspect traffic from ports and block traffic
* Need a signature of what they should block (difficult)
* If malware is sending traffic that’s encrypted, then it couldn’t be decrypted unless there was a certificate installed on the browser

Host-Based firewalls

Installed on the device you use

* Komodo (windows)
* Iptables (Debian/Linux)
* PF Firewall and application firewall

Disadvantage as being on the same platform/device as the malware

Because of this, the malware on the device could just disable the firewall

As well as using trusted processes, such a s the browser, to communicate out and bypass the firewall

An advantage is that they can understand what applications are running so they can block applications based on a black-list and a white-list

* This way malware couldn’t make outbound connections because its not in the list of trusted code
* But again, malware that knows what it’s doing could bypass this by riding on a trusted process like the browser

Virtual Firewall

Could have a PfSense VM that you route all other VMs through to protect your other VMs

With VMS

* Adapter 1 should be connected to NAT
* Adapter 2 should be connected to your virtual firewall e.g., PfSense

General firewall rules

* All network traffic should be denied unless implicitly allowed
* Specifics to block
  + IPv6 if not in use
  + UPnP:1900
  + IGMP
  + Any Mac, Windows or Linux service that is not in use
* Your firewall should be set up specifically for your set of rules and landscape