14 Security and Trust Part 2

In this section, we'll look at some of the standard ways in which we protect networks and look at some attacks on them.

14.1 Classic Networking Protection

Network security is typically a balance between safety and ease of use.

The classic view of network security is analogous to a castle.

- You have strong borders that prevent anything passing through.
- You have specific controlled entrances that are heavily guarded with checks so that only "good" things can go in and out.
- Inside the castle, security is more relaxed.
- Depth of security is provided by additional layers of walls/entrances to create safer parts and to ensure that a breach in one area cannot necessarily spread.

When applying this to networking, we end up dividing the world into three types of area.

- **Inside / Trusted**. This is a safe zone, where endpoints are assumed not to be malicious. That doesn't mean no checks, but typically not many. For example, consider a corporate or university internal network. As a user you must authenticate yourself when you log on, and there are typically some restrictions, but you can see other machines in the network, log in log in to them, move files around, etc. without having to go through extensive network checks. Connections are typically not restricted *out* of the trusted network, but connections in are restricted.
- **Outside / Untrusted**. The rest of the world (for example, the internet). Outside endpoints are assumed to be malicious until proven otherwise and incoming connections to other areas are carefully policed.
- **Demilitarized Zone (DMZ)**. Borrowing terminology from the military, a DMZ is an in-between area. Within a DMZ, there are untrusted endpoints operating, but connections within the DMZ are allowed. A good example is an interoperability test lab. You might allow third parties that you don't trust much to install and run their endpoints in the lab network and communicate freely with other endpoints within the lab – but you very carefully police traffic away from that lab to your trusted network.

14.2 Borders

14.2.1 Air Gapping

The simplest and strongest method of preventing information travel between two networks is to have two physically separated networks. Having no network connectivity (including no wireless connectivity!) is called air gapping, and provides extreme network security and no network usability.

In reality, you might provide separate networks using VLANs rather than physically separate networks.

Importantly, note that *extreme* network security is not the same as complete network security. Other methods of getting information in and out of a completely air gapped network include USB sticks, soldering or otherwise installing additional secret (typically wireless) networking hardware wireless chips inside kit in the network, or even Van Eck phreaking. These are all beyond this course, but the reader will probably find an internet search for Van Eck phreaking interesting and illuminating.

14.3 Controlled entrances

14.3.1 Firewalls

A firewall is a set of filtering rules on a networking relay (typically a router) that sits on the border between the zones of trust.

These filters typically place rules on the source or destination address or ports that are allowed to make connections, but can be more complex (allowing or preventing different types of traffic, different protocols, etc.) Filters can be static (permanent) or dynamic (dependent on recent traffic). By default, firewalls block requiring the user to explicitly open up holes. Some examples of filtering rules:

- Block all incoming connections unless they are to port 80 (where our external facing website is listening for incoming traffic).
- Block anything coming from 8.16.0.0/16 (where we've had lots of malicious attacks in the past).
- Block all incoming traffic unless it is a response to a connection that has gone out in the last 30 seconds.

These basic filters must be fast and efficient in order to prevent denial of service attacks where a malicious user is sending traffic blocked by your firewall and it spends all of its resources blocking and dropping traffic.

14.3.2 Network Address Translation

Firewalls are often used in combination with Network Address Translation.

As you'll remember from earlier in the course, NATs allow networks to use private address spaces – though that is not explicitly required. More importantly, internal connections outwards cause the NAT to create dynamic mappings between address and port on the inside and with address and port on the outside of the NAT. Externally-initiated connections need to have a permanent pre-configured mapping in order to work.

This fits perfectly with firewalling, as by default nothing outside can initiate connections into the trusted network – and in fact nothing outside is even able to see the addresses of endpoints inside the NAT (as the NAT replaces the IP addresses with its own!)

14.3.3 More complex firewalling

The above filter rules described provide filtering based on network or transport layer flows, but what about applications where you might want to filter on the basis of URLs or other contents of a packet. Accessing the internals of packets beyond the network headers is called **deep packet inspection** and requires the firewall to have application specific knowledge.

Example: Session Border Controller

A session border controller provides NAT/Firewall protection for Voice-over-IP networks. Phone calls are set up using SIP (Session Initiation Protocol) which runs over TCP/UDP/TLS and the SIP protocol itself is used to negotiate addresses and ports (and media encodings) used for the later media connections for the call. The SBC acting as a NAT must both handle translations of the SIP URLs *and* modify the media addresses/ports inside the protocol to make sure that both endpoints send media flows to it (rather than trying to talk directly which will fail). Finally it also needs to dynamically program specific holes in the firewall rules for the media to flow through (and close them down again once the call has finished).

14.4 Virtualisation attack vectors

Classic networking protection is useful (security consultants will disagree *how* useful) but does not provide complete protection in virtual networks.

VLANs / VPNs

With VLANs and VPNs your networks are not physically separated from others, so you must assume that everything is "outside" or at least visible from "outside". For example, encrypt your internal traffic and use authentication on your internal infrastructure.

Virtual Machines

With virtual machines, networks are now no longer needed at *all* to traverse between endpoints. It is possible to attack both from one VM to a neighbouring VM on the same physical host, or indeed to attack from the VM to the host itself. These attacks operate as programs escalating privileges to reach/access other programs on the same hardware than "networking", and details here and the protections against them lie outside the scope of this course.

14.5 Attacking and cracking networks

This is a networks course, not a security course. However, understanding network security and the problems with it in real life is very important when you come to create and set up networks. Good knowledge here is critical to being able to secure and defend your networks.

Here is an initial list of possible attacks and defences that you could learn about. Nothing from this point on is examinable.

3 Starting points

- **Password cracking.** Checking for common passwords and dictionary attacking passwords are simple but effective. At the very least making sure that anything accessible externally has some kind of small scale pause between connection attempts from the same place helps slow these down.
- **Port scans.** A port scan is simply attempting to connect to each port in turn on a device to see what is open, what is listening and what responds – looking for a way in to a poorly configured host.
- **Buffer-overflow attacks.** A buffer-overflow attack is simply when you read or write beyond the end of a buffer that you have access to. This is a reasonably common type of attack in networking (See Heartbleed example below).

Kali Linux

Kali Linux is a distribution of Linux that comes with a large selection of useful and interesting tools, including various networking scanners and port scanners. If you are interested in learning more about networking security in order to help effectively secure your network, I recommend downloading it and giving it a play.

DefCon

DefCons are meetups for folks who are interested in aspects of computer security with both talks and networking/discussion afterwards. Local groups are named after the local area phone code. For example, the London Group is DC4420 (+44 for UK and 020 for London). DefCon folks are usually helpful and welcoming to newbies – but don't take any unsecured electronic devices with you…

14.5.1 Examples of Attacks

Heartbleed

Found in 2014, Heartbleed is a standard buffer-overflow attack in one of the most common SSL/TLS libraries – OpenSSL- which at the time was used by just about everything. The library included heartbeat messages to check the liveness of a connection. The implementation here was that the client sent the server a request with a string of text (with the length of the text also provided in the request) and the server responded with that same string. The server code was missing length checking, so a client could provide a short string of text but supply a false (very large) length for the string. The server's reply overread the string stored from the request and provided the contents of the surrounding memory. That surrounding memory was being used by the SSL/TLS library, so was full of useful keys, passwords, etc.

Ping of Death

Ping of Death is a straightforward attack on routers using IP fragmentation. There is a maximum length of an IP packet that can be created, so routers don't need to cope with receiving a larger one. However, if you manually create a set of fragmented packet pieces that *when you recombine them add up to a larger-than-you-could-feasibly-have-created max sized packet,* then the receiving application may fail. Many applications expect packets under a certain size, and crash if they receive an oversized packet.

This is called *Ping* of death, because there were a number of vulnerable "ping" applications on routers that crashed on receipt of a mega-sized ping packet. (ICMP ping packets are default <100 bytes so dealing with theoretical impossibly large packets wasn't thought about by whoever was writing that code at the time).

VENOM

A Backronym of Virtualised Environment Neglected Operations Manipulation, VENOM is a buffer-overflow attack from VM into the host. It was present from 2004-2014 versions of QEMU, found in Xen, KVM and VirtualBox hypervisors. A subtle interaction between 2 of the handling functions in the QEMU virtual floppy disc controller allowed an attacker to overwrite off the end of a VM's buffer and into host memory.

14.5.2 A final word of caution.

If you want to learn more about any of these things, please go ahead. If you want to have a play around with tools to help you learn how to secure your network, or if you want to experiment to understand how easy it is to knock out network infrastructure *on your own personal network and devices*, to better understand things, I can't stop you. But be careful on two counts.

- ISPs and network administrators are understandably allergic to people messing with their stuff and the law is almost always on their side even if you accidentally do something really minor that you didn't intend and didn't cause damage. In fact depending on where you are, the law might not allow you to do things even to your own devices.
- Many folks who work in this area have norms (especially technical privacy norms) that are different from what you might consider normal or acceptable. For example, if you go to DefCon and you have Bluetooth turned on, expect at the very least people wandering around inside your device having a look at what you've left out on display.