Case 1

De stand-alone pc en het internet

Bart Janssens
PCE

Index

This page is intentionally left blank.

The OSI model

History

The OSI model was designed by the ISO (International Organization for Standardization). This was done to make it possible for different systems to be able to communicate with each other. As in the early days of computing this wasn't that easy. In the early days you usually bought a system from a vendor and then had to buy all the other things you needed from the same vendor. As the OSI model was developed in a way to set a standard that every system could use, made that the buyer was no longer depended on the vendors.

It was made as a guideline for how network communication should take place. Early experience with the forefathers of the Internet as we now know it lead to a layered approach. This made it possible to change a part of the system without making other parts fail, they are independent of each other. As each layer does a specific task and only that. There is of course the issue of the intercommunication between layers.

So what are the advantages to this model? As first: by dividing a big thing in smaller things makes it easier to solve problems. As there is a layered structure the vendors can write to a common in- and output. This makes their application communicate to a common thing instead of all of them. Also the layers are not depending on each other, meaning that changes in one layer does not affect the others. The devision in layers makes it easier to standardize functions. And of course it works with different types of hardware and software. Another used model for communication is the TCP/IP model that only has 5 layers.

Introduction

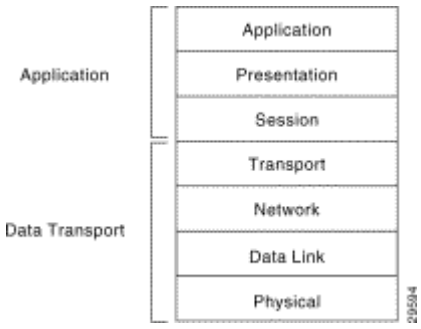
The Open Systems Interconnection (OSI) model describes how information from a software application in one computer moves through a network medium to a software application in another computer. It is made up from 7 layers that specify particular network functions. The model divides the tasks that are needed to move information in the network into seven smaller task groups.

A task or group of tasks is then put into one of the seven layers. Each layer is self-contained so that the tasks assigned to that layer can be implemented independently. This makes it so that the the messages provided by one layer to be changed without affecting the other layers.

The 7 layers of the model

The OSI model from ISO has 7 layers. These make up the model. The model is in essence a framework for communication between computers, the model itself is not a method of communication. This is made possible by the communication protocols. An easy way of remembering the structure is the following: "All People Seem To Need Data Processing". This it a common way for students to remember the sequence of the layers:

7 Application
6 Presentation
5 Session
4 Transport
3 Network
2 Data link
1 Physical



Physical layer

The first layer defines the physical link between communicating network systems. It's about the electrical, mechanical and procedural specifications for activating, maintaining and deactivating the link. The physical layer handles the voltage levels, timing of them, data rates, transmission distances and physical connectors. They can be categorized as either LAN or WAN specifications.

Data-link layer

The data-link layer ensures a reliable transit of data across the physical network link. It also defines differed network and protocol characteristics. These include physical addressing, network topology, error detection, sequencing of frames and flow control. Physical addressing is the way how devices at the physical layer are given an address. Network topology is how physical devices are physically connected.

Error detection alerts upper layers about failing transmission and sequencing of data frames requests by resending frames that are transmitted out of sequence. Flow control moderates the transmission of data, meaning that it will manage the speed of the connection if the receiving device is overwhelmed.

The layer is subdivided by the IEEE into 2 sub-layers: the Logical Link Control (LLC) and Media Access Control (MAC). The LLC layer manages communication over a single link of a network. It supports both connectionless and connection-oriented services used by the higher layers. The MAC layer manages protocol access to the network medium, it also specifies a MAC address which can be used to uniquely identify devices at the Data-link layer.

Network layer

The third layer allows for multiple data-links to be combined into an internetwork. It also provides routing. This is done by using logical addressing of a device. The layer supports both connection-oriented and connectionless services from higher layers.

The network layer protocols are typically routing protocols but there are also other types of protocols implemented in this layer. These other protocols are tunneling and some vendor specific ones like IBM´s SNA.

Transport layer

The transport layer allows for reliable internetwork data transport services that can be used by the upper layers. Typical functions are flow control, multiplexing, virtual circuit managing, error checking and recovery. Flow control is managing the transmission rate between devices. Multiplexing enables that data from several applications can be transmitted onto a single physical link. Error checking involves detecting transmission errors while error recovery is resolving any errors that have occurred.

Session layer

This layer establishes, manages and terminates communication sessions between upper layer entities. It is responsible for the graceful close of sessions.

Presentation layer

The sixth layer in the OSI model provides conversion functions. These are used to ensure that the data transmitted through the other layers will be readable by the application layer of a different device. These can include conversion schemes and common data representation formats, but also

common data compression and encryption schemes.
The use of standard representation formats are applied in this layer. These are essential to make that the data can be used on different devices and by different applications. Also that encrypted and compressed data can be decrypted and decompressed.

Application layer

The last layer is the closest one to the end-user. This means that both the ISO application layer and the user directly can interact with the application. This layer interacts with the software applications that implement a communicating component. Such applications are not included in the scope of the OSI model.

8th Layer

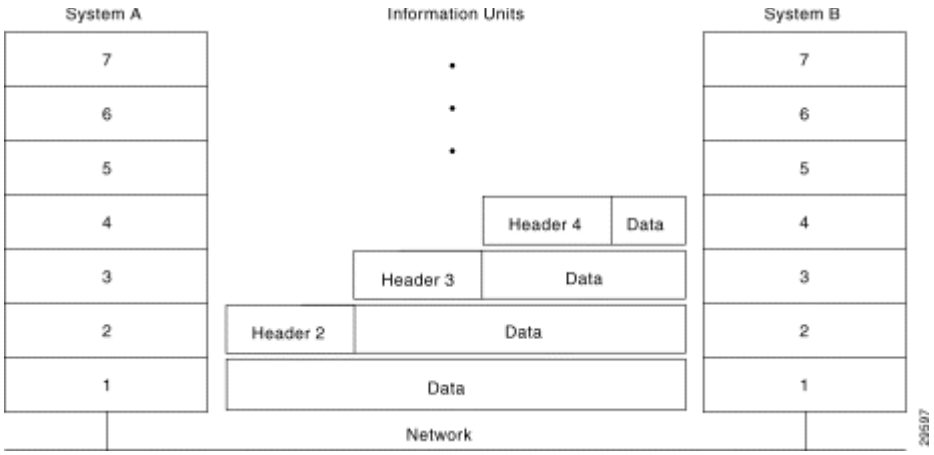
The 8th layer is the person that operates the computer. This is the place where the most errors happened. This is know to network engineers as pebkac: Problem Existed Between Keyboard and Chair, meaning the user.

Protocols in the model

List of which protocol lives in which layer can be found on the Internet so they will not be listed here to make up pages. This will be done by clever use of fonts and layout.

A protocol, in a networking context, is a formal set of rules and conventions that governs how computers exchange information over a network medium. A protocol implements the functions of one or more layers. The protocols that are defined in the OSI model where later adapted to the TCP/IP model.

Information that needs to travel from one end-application to another one on a different device will be handed down through the layers and send over a physical link. Then the information will travel back up through the mode until it arrives at the end-application. As the different protocols handle the data they will add headers and sometimes trailers. This process is know as encapsulation. Theres is a need for this as the receiving layer will need to be told what to do with the data it gets as it is traveling back up through the layers. As the protocols remove the headers and trailers the data makes its way back up in the stack, this is called de-capsulation.



Connections

Networking protocols and the flow of data can be separated into two types of connections: connection-oriented or connectionless.

Connection-oriented service is made up of three steps: connection establishment, data transfer and connection termination. During the connection establishment phase a single path is choice between two devices and the network resources are reserved to ensure a consistent service. Then, in the data-transfer phase, data is transferred sequentially over the single path. Data arrives in the order it is send. After the data-transmission phase, the connection termination phase is initiated, as the established connection is no longer needed the connection is terminated. If there is again a need to transfer data, this will require for a new connection to be established.
Connection-oriented network service has two disadvantages over connectionless service. The static path makes it so that if somewhere among the path there is a failure that the connection will fail. Also the reserved resources cannot be shared among users, meaning that if the throughput rate isn't maximum, bandwidth is wasted and that makes for an inefficient use of resources. They are useful for transmitting data from applications that do not tolerate delays and packet re-sequencing, like voice and video services.

Disadvantages of connectionless service is that each packed needs to be fully addressed and is handled independent. As there is no predetermined path where the packages are getting send. However it provides dynamic path selection and dynamic bandwidth allocation. This allows for traffic to be rerouted through failing paths. The bandwidth usage is more efficient as network resources are not reserved.

IP Addresses

An IP address is a number used to identify nodes in a network. This number can change according to the DHCP and DNS settings in the network. It can also be static.
There are 2 types of IP addressing methods currently used, ipv4 and ipv6. This is a outcome of there being not enough ipv4 addresses to accommodate all the nodes in the network. Another solution to this problem is sub-netting.

The structure of IPv4

IPv4 consists of 32 bits, these can be divided into four octets. This means that the structure looks like this:
00000000.00000000.00000000.00000000 (the dot '.' is to make the address more readable for humans)
Each octet can then be converted into a decimal number that is in range between 0 and 255. The address also gets divided into a host and network portion. This is done to provide an addressing scheme that can accommodate variable sizes of networks. The network part start at the most left bit of the address, and the host part is the most right part of the address.

Classes

There are 6 classes that are defined to be used. They range between A - E although in the practical world only A, B and C are used as D is used for multi-casting and E is for experimental use.
The classes can be identified by the network part of the IP address as shown below:

```
ADDRESS                                  CLASS       RANGE      CIDR

0xxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx Class A     1 - 126    /8
|network|      24-bit node-id      |

10xxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx Class B   128 - 191    /16
|      network    |  16-bit node-id |

110xxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx Class C   192 - 223    /24
|          network          |  8-bit |

1110xxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx Class D   224 - 239    N/A

11110xxx.xxxxxxxx.xxxxxxxx.xxxxxxxx Class E   240 - 255    N/A
```

In the A class only the first octet is the network portion, this increases to the first 2 octets in the B class and in the C class only the last octet for the node-id. The node-id part of the address can then be divided by the network manager to divide into subnets. There are some special cases like the 0 and 127, these have a special function to them. The 0 is the default gateway and 127 is the loopback so all the addresses range 127.0.0.0/8 refere to the same address, the localhost. CIDR stands for Classless inter-domain routing, and it is used to replace the design based on classes. It allows for non-octet boundaries.

Sub-netting

Sub netting is used to divide a class into multiple networks. This was done to accommodate the fast growing network. Sub-netting is just extending the original CIDR and so 'borrowing' bits from the host portion of the IP address to the network part. This makes it possible to have other CIDR's like / 26, this would be a class C borrowing 2 bits:

```
X. X. X. xx| 000000
|network|  host |
```

This would allow for 63 (1+2+4+8+16+32) hosts on the network as one address (X.X.X.192) is the network ID and X.X.X.255 is the broadcast. So the range is X.X.X.193 - X.X.X.254.

IPv6

The IPv6 protocol was designed to replace the current IPv4. This because of the limited supply of IPv4 addresses. As it uses 128-bit long addresses there are approximately 3.4*10^38 addresses available. The replacement of IPv4 with IPv6 many cause some issues as the two protocols are not compatible with each other, there are conversion algorithms to solve this problem. IPv6 consists of 128 bits or 8 groups of 4 hexadecimal characters separated by a colon.

DNS

Introduction

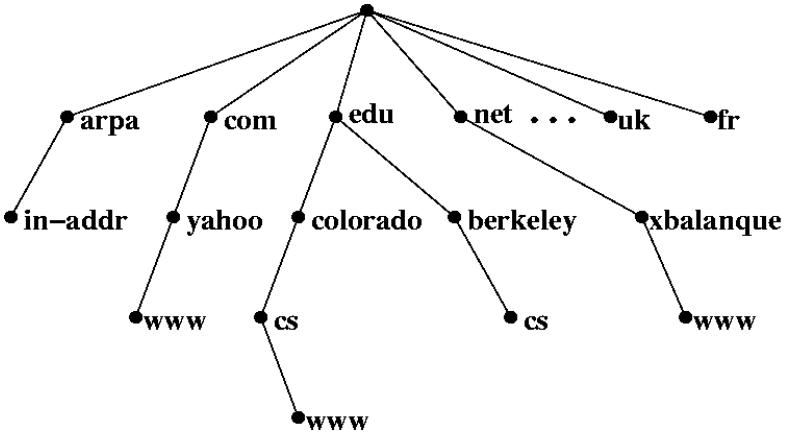
DNS stands for Dynamic Name System, it enables the translation of an IP address to a host name. This was done in order to solve problems that were arising as ARPanet was made public.
As it is easier for a person to remember a name, as that is descriptive of the service it provides, there was a need to be able to translate the name to the IP address. This was first done by maintaining a list that had all the names and IP addresses of all the host. As the number of host grew, the list became longer and the lookups preformed took longer. This was an issue as the user needed to wait longer and queues began to from. This resulted in users getting a time-out. The time-out problem was initially fixed by using secondary and thirty name servers. This spawned another problem. As the list was maintained manual by the administrators and this made it an untrusted resource as the admin needed to update all the name servers. There was need for a system that allowed organization in a hierarchical fashion, that is scalable and that is easy to manage. Enter DNS.

DNS

DNS is build on top of three principles:
- It must be hierarchical,
- It must be able to spread the load,
- It must delegate the administration.

Hierarchical

DNS uses a hierarchical system. This means that it is build like a tree, with the top of the tree being the root '.' . Next are the Top-Level Domains (TLDs) and then there are the Second-Level Domains(SLDs) followed by any number of lower levels, separated by a dot. So for example: graphitedev.dev.inuits.eu, eu is the TLD, inuits is the SLD, dev is a lower level and graphitedev is the host name of the node in the network.
TLDs consist of two types:
- Generic TLDs, these are: .com, .org, ...
- Country Code TLDs, these are: .be, .se, …



Delegation

The domain authority and delegation are very important in the DNS system. Each node within the network is assigned to and authority, someone that is responsible for the management of that node. The authority for that node can then delegate the authority for the levels under that node within the hierarchy. So lies the authority for the root with ICANN. It in turn delegates. This way the admin that administrators the node needs to manage the dns record of the node, if he places it inside his own network of course.

Lookups

Dns lookups happens when a local dns server doesn't know the answer to a query. In this case it will ask a name server in the level above it for an answer. This process will repeat until an answer is found. Then the answer will be returned to the node that queried for it. This means that the database is distributed. As there isn't a single server that has all the answers.
This is done to spread the load on the servers.

Registration

If you want to register a domain name, you will need to contact the ICANN organisation. They will ask you to submit your personal contact information to the WHOIS database. Once you do that and ICANN has entered you into the list, then your address will be publicly available. Your entry will have an expire date, so if you want to keep your domain name you will have to renew it.

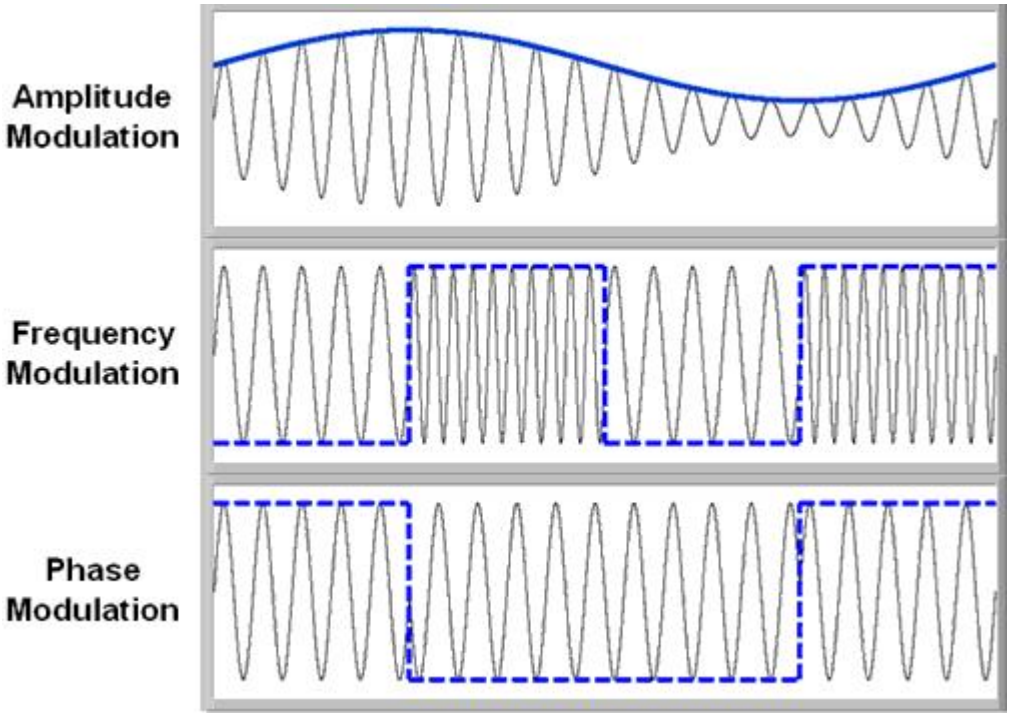
MODEM

Introduction

The modem is a piece of hardware that allows the translation between analogue and digital signals. This is used by the ISPN to connect you to the internet. It makes use of existing copper cables that are also used for telephony.
Modems are classified by using how much translating they can do in a time unit, aka mB per s or kB per s.
Multiple standards were developed to do this modulating.

Translation

A modem 'translates' a digital signal to an analogue one and the other way around. This is done by modulation of the signal. The modulation of a signal is done by using the amplitude of the signal. By mapping different binary values to the amplitude of the signal it can carry a 0 and 1.
Is was of course not very efficient as the signal can take more as two values for the amplitude. By adding multiple bit together a higher rate could be achieved. This is called multiple amplitude modulation. Next to the amplitude of the signal, the frequently can change as well. This is called phase modulation. This is done by mapping different frequencies to bit combinations.



Standards

Multiple standards were developed to do the modulation. The two most common ones are ISDN and ADSL, respectfully Integrated Services Digital Network and Asymmetric Digital Subscriber Line.
ISDN integrates both speech and data on the same line. The speed on the line was 128kbitps as it used two channels of 64kbitps.
ADSL uses frequencies that cannot be used for speech for data. This enables the use of speech and data simultaneously.

Administration of modem and router

As most ISPs now integrate the modem and router this chapter will discuses them both together. It will also discus only the Belgacom one assuming that other routers will be similar.

Belgacom

The ISP Belgacom (also know as Proximus) delivers a b-box modem/router. This bbox is in the 3th generation now. They claim that it is more powerful and more efficient but also easier to use. As I have only the b-box 2 I will focus on that model. These modems that the company uses are actual a modem and a router together.

Setting up

The first thing to do is plug in all the cables that come with the modem, depending on what kind of subscription you have the configuration changes. Belgacom offers a complete pack with internet, telephony and tv included. Therefore it is logical that the user will need to have more and other cables plugged than if the user only uses it for internet. As this is the only use as my home and the others fall outside the scope of this task, it will be the only configuration Ill disgust.

There are several ways to have an incoming connection from the provider:
- one 5-point contact
- one 6-point contact
- multiple contacts

One 5-point contact

Begin with removing the plug, now plug in the adapter and then the VDSL2 splitter. Next use the VDSL2-cable to connect the bbox and the splitter. Now connect the bbox to power. As the bbox is now powered the on button should light up, if this is not the case then reconnect the power cable to the bbox. If the DSL starts blinking the bbox is configuring itself. this should take around 2 minutes. Once the on, DSL and the wireless leds start blinking the bbox is ready to use.

One 6-point contact

Plugin the VDSL2 splitter, follow the steps above.

Multiple connections

First unplug all the contacts, then find the first contact. This is done by systematically disconnecting all the sockets till there is only on left with a dail-tone. Use the first point and repeat all the above steps.

Internet connection

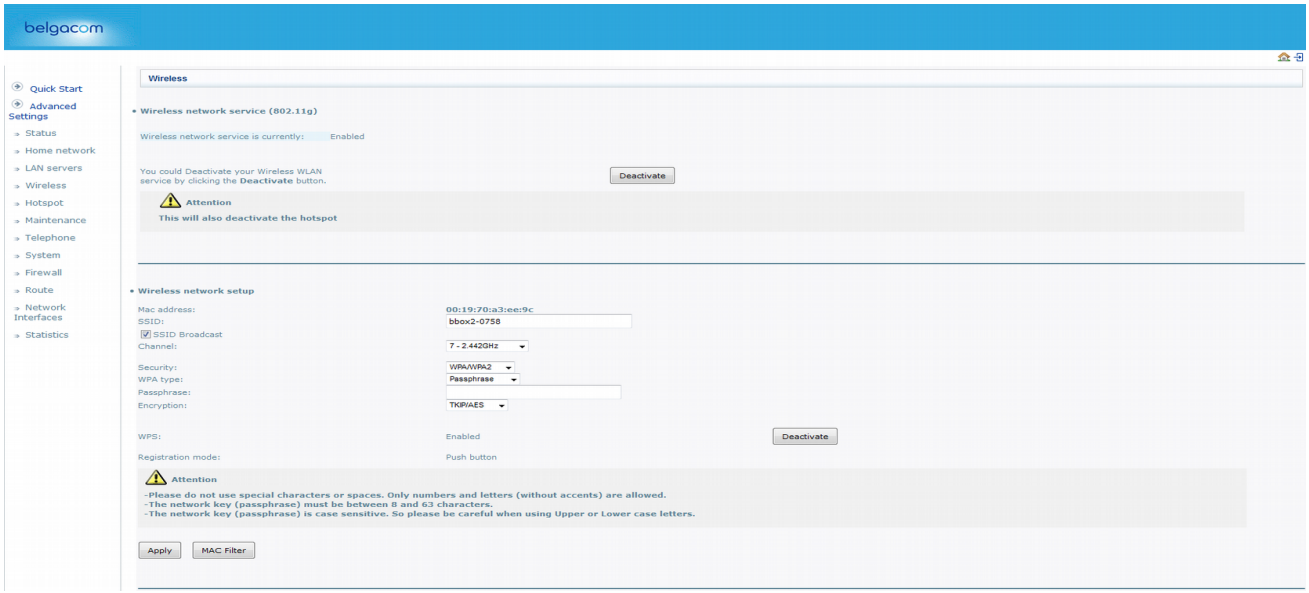
If the @ led on the bbox is burning, it means that the modem automatically is configured. More specific that the login and password automatically where set-up and don’t need to be given to use the internet. Now, using the Ethernet cable, connect the bbox to the pc. If the connection is active, the LAN leds will start blinking. To see if there is connection to the internet go to a website and see if it loads.

Wi-Fi connection

To set-up the Wi-Fi connection point your browser to 192.168.1.1 . Were provide the Serial Number of your bbox. This can be found on the bottom of the bbox. If the given SN is correct it will take you to the status page of the bbox. This page gives a quick overview of the connections. To set-up the wifi co to Advanced Settings and then Wireless. Here the SSID, Channel, Security, WPA type, Key and Encryption can be configured. On the top of the page there is the button to enable and disable the wireless connection. At the bottom of the page there is the option to enable the MAC Filter, this only lets the devices with a certain MAC Address join the network.

- SSID: the 'name' of the network, use a name that is unique so you can easily identify it.
- SSID Broadcast cheekbone: This makes the SSID visible to everyone.
- Channel: this is the frequency where the wireless signal will use, if there are signal strength problems, change this and test the connection. Signal strength issues are mostly caused by interference of other routers.
- Security: this is to set the encryption type, select WPA/WPA2 to have the most secure.
- WPA type: lets you choose the way to authenticate to the network.
- Pass phrase: enter the password you want to use to authenticate.
- Encryption: if the Security is set to WPA use TKIP, if it is WPA2 use AES.

Click the apply button to use the configuration. Now the wireless can be used with the SSID and the



Advanced settings

Under the advanced settings there are a number of sub-menus. From top to bottom there are:

- Status
- Home Network
- LAN servers
- Wireless
- Hotspot
- Maintenance
- Telephone
- System
- Firewall
- Route
- Network Interfaces
- Statistics

Status

This tab shows the status of the bbox. It contains information about the Internet connection and the Gateway, the wireless connections and shows the logs fro the bbox and the DHCP server. There is also a Vlan list, this shows the connections to the internet and the connection that is used for firmware updates.

Home Network

The Home Network tab show the settings for the internal network configuration. These are the IP address of the bbox and the Subnet mask. It also shows the configuration of the DHCP server.

LAN Servers

This tab shows the port forwarding on the bbox. This page consists of a table that shows the name, protocol, port range, and the Local IP address. This is where the port forwarding is configured. The user can add a port forwarding rule by clicking New Entry and inputting all the needed information.

Hotspot

On the Hotspot tab the user can enable and disable the Belgacom hotspot, it also gives information on the status of the hotspot.

Maintenance

You would expect a big button that says: "Check for firmware update" but sadly there are only a "Restore to factory default configuration" and a "Reboot" button.

Telephone

The Telephone tab show all the configuration there is to the telephone setup. This can be viewed for the different Lines the user has or in general. There is also a tab that shows the call history.

System

This tab shows a list of users on the system, still no "Check for firmware update" button.

Firewall

Here is where all the firewall configuration is done. In the General tab there is the option to choose between default settings on the level of security. The Access Control tab gives the ability to block internet access from within the LAN. So you can prevent a device with a certain address in your LAN to connect to the internet. The DMZ Host tab does the opposite of the Access Control one, it allows for a device to be fully exposed to the internet. The Port Triggering tab is used to set-up which ports are open to the internet. These can be specified by protocol. Under the Remote Administration tab is a list of check boxes that can be used to open ports to allow remote administration. This can be port 23 for telnet or port 80 for HTTP access. Under the Website Restriction tab is the option to block certain websites. This means that for a device in the LAN network you can block access to, for example, facebook.com. The advanced filtering enables the user to setup rules for package management, what to do with packets from certain hosts, etc. Finally the Security Log tab shows, well..., the security log.

Route

This tab show the configuration of the DHCP and DNS.

Network Interfaces

This tab shows a list of the available network interfaces on the bbox.

Compression

The need to compress files was born out of necessity as the old connection to the internet only could carry so mush bits. Therefore a way as invented to make the files that were send over the connection smaller. There are several used compression algorithm but they all relay on the same principle:
"To use fewer bits than the original representation."
There are two types of compression, a lossy and lossless and as the names might suggest the lossless doesn't remove any information and the lossy does. As the lossless is based on statistics and the lossy on the removal of information that cannot be perceived by humans. Example: There is an image. The lossy will remove all the colours that cannot be viewed by a human so every wavelength that falls outside the 380nm and the 750nm. The lossless will analyse the image bit per bit and look for repeating patterns. If there is a whole sequence of bits that present the colour blue, eg. the sky, it will say something like: "from bit 100 to 1500, the colour is blue". This will take less space then saying: "This bit is blue" a 1400 times.

Encryption

Introduction

Ever since there was a need to secure information there was a need for some sort of mechanism to 'lock' data. So that only the parties with the correct 'key' could access the data. The Romans used pigeons to deliver and receive orders and reports on enemy movement. It was important that the enemy couldn't get these reports as they could contain information on tactics or possibly expose weaknesses in the strategy. Therefore the Romans used a system to convert the messages into what seemed a random string of letters and numbers. Of course there was the need for a 'key' to deciper the message. And there lies the weakness of this kind of 'trusted key' mechanism, if the enemy gets their hand on the key, the system fails.

We have come a long time since the Romans and their pigeons. With the wars the have been fought over the years the secrecy of information grew more important and now that the internet can be found in every home the safekeeping of data should be heigh on the list of important things. With the recent 'Fappening', where pictures of celebrities leaked on the internet, people should realize how important data is and that they should do an effort on securing it.

Different kinds of encryption

Symmetric key

This type of encryption is similar to the Romans way, both parties need to uses a key to encrypt and decrypt the data. This has the same drawbacks as the Romans also, both parties need access to the key.

Asymmetric key

Also known as Public key encryption, this mechanism relies on two different keys. One of them is a secret or private key, the other is the public key. These keys are generated together and so they are linked to each other. This link is important regarding the different function of the keys. The public key is used to encrypt the message. Then the private one is used to decrypt the message. Seems like the symmetric key system right? No, as the message encrypted by the public key can only be decrypted by the private key. So this ensures confidentiality as only the one with the private key can decrypt the data.

The use of this system also allows the signing of data. Meaning that if a message is signed with the users private key, it can be controlled on integrity by anyone who has the public key. This is a common sight with package management as the repository asks for Pretty Good Privacy key or a PGP. This key can then be used to see if the packages are the correct ones and not some package made by someone with bad intentions.

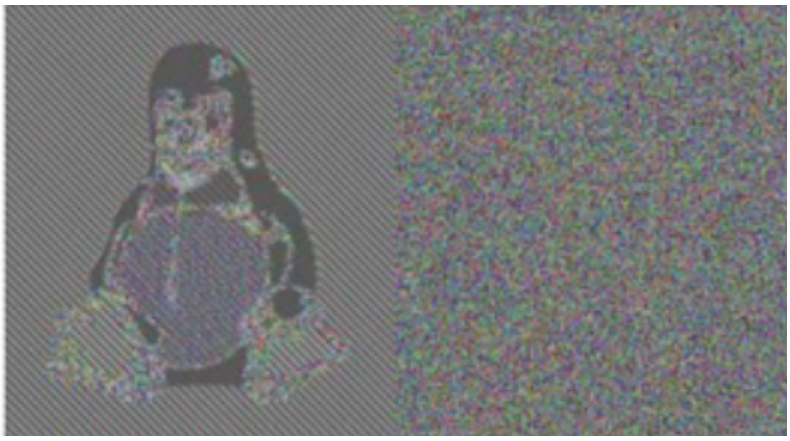
Hashing

This isn't really encryption but is more like a one way thing. Hashing is mostly used to store passwords from users in a database. Generating a hash is a mathematically irreversible processes. This means that even if a hacker gets access to the database containing the passwords of the users, he is unable to get them. Hashing generates a string of characters form a given input. This string is then saved. As the users returns and enters his/hers password the hashing algorithm will generate the string again and compare it to the one in the database. If these don't match it knows that you have given the wrong password and will revoke access. This is why if you forget your password they send you a link to change it as they actually don’t know it.

Another use for hashing is checking if a file has been tamped with. If the up loader puts up the hash to the file, the one that downloads it can hash the file and compare the hashes.

Problems with encryption

There where some problems to solve with encryption. As the algorithm needs to produce a random string that doesn't give any data on the message. So if the data is “This_is_a_secret.” uses bad encryption it could give away the length of the words in the message. This problem gets worse as the length of the message increases. There is a good example with an image:



On the left there is the Linux penguin. As we use an algorithm with a low enthalpy, the penguin reappears even in the encrypted file. If we use an algorithm with a high enthalpy, the image gets truly random with no reappearing sections.

Error handling

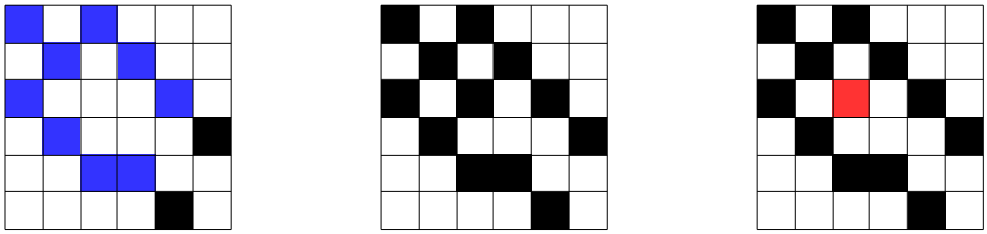
The general idea behind error detection is to check if the transmission of data was successful or not. This is important as the upper layers of the OSI model expect data without errors in it as some applications don't behave as expected or stop working completely as they receive faulty data. This is why error detection and correction is handled by the lower data-link layer.

There are 3 types of errors that could occur:
- Single bit error
- Multiple bit error
- Burst error, when there are multiple sequential bits wrong

There are two ways to handle errors:
- Error detection
- Error correction

Error detection

With error detection the data is divided info frames. To these frames extra information get added on how the data is structured. For example:



The data is represented by the 5 by 5 grid. Then we add another row and column. We fill up the squares so that in each row and column the amount of black squares is even. This way we can detect if there was an error. This is also know as a parity check. This kind of technique only works with single bit errors.

Some kinds of error detection schemas are very inefficient as they just send the data multiple times. This allows for the data to be check but is heavy on the network. The most commonly used mechanism for check data integrity is hashing.

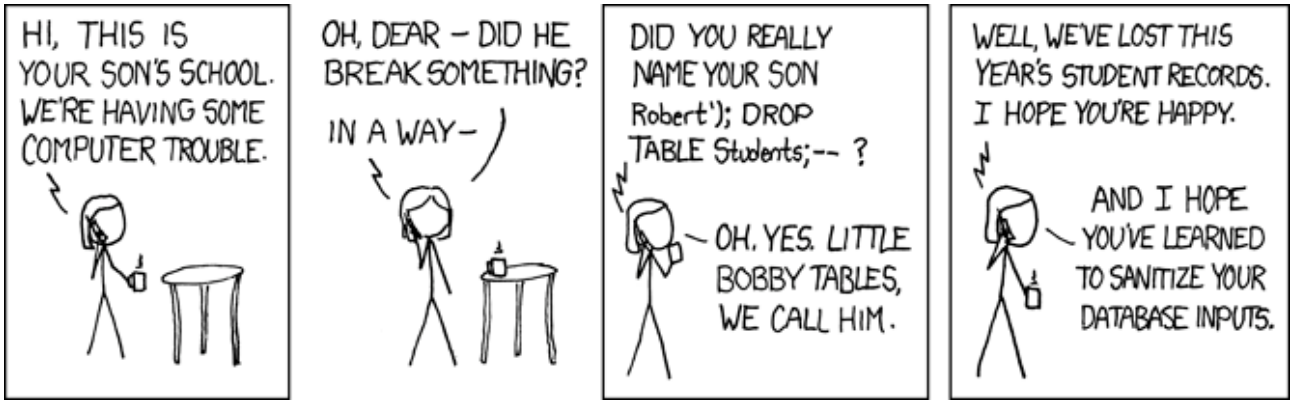
Error correction

Error correction is done in two ways, either there is a request for another copy of the data or error correction is done. This allows allows for recovering of some errors. This is done on a bit-by-bit or block-by-block basis. The block-by-block method is used in the bit-torrent-protocol. As the entire file gets split into blocks. These blocks get hashed and the hash and the block are transmitted. If the block doesn't give the same hash the bit-torrent client knows that it needs to request that block again and not the entire file.

Viruses

In the early days of the internet viruses and malware where somewhat of an exploitation of bugs, found by users and then used for personal entertainment, demonstration of the bug, sending a message or for personal gain. Its the last use that is the most concerning. Also viruses doing damage to systems is a huge problem as it can result in the waste of computer resources. This ups the cost of running the machine to do the original tasks and ups the maintenance. Therefore a new industry has risen, software companies who develop anti-virus programs.

The most recent viruses that have come to light are the ones that are caused by bad coding. Take the heart bleed bug. This bug was in the code but due to the way that code was written it was never found. As a result a very large part of the internet was vulnerable. Another example is the recent bash bug where a command could be executed by exploiting a system variable. This was actually a typical injection type of bug. Comparable to sql-injection where the user gives input and ends it with a ';'. This makes the database think that it is the end of the input and will try to execute the command behind it.



These kinds of bugs affected all kinds of systems. But there are also very operating system specific viruses. As the windows operating system has such a big market share it gets most targeted by malware and viruses. A new kind of this is the cryptolockers. Once these infect a system they encrypt everything and ask for money with the promise that in return they'll send you the code that will decrypt everything again.

Sources

https://tools.ietf.org/html/rfc3439

http://tools.ietf.org/html/rfc1122

http://www.cisco.com/cpress/home/home.htm

http://www.cisco.com/cpress/cc/td/cpress/fund/ith/ith01gb.htm

http://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html

http://tools.ietf.org/html/rfc2317

http://www.faqs.org/rfcs/rfc1817.html

http://tools.ietf.org/html/rfc5735

Unix and Linux System Administration Handbook, ISBN: 9780131480056

http://www.tutorialspoint.com/data_communication_computer_network/error_detection_and_correction.htm

http://www.youtube.com/watch?v=vCDe14NxSY0