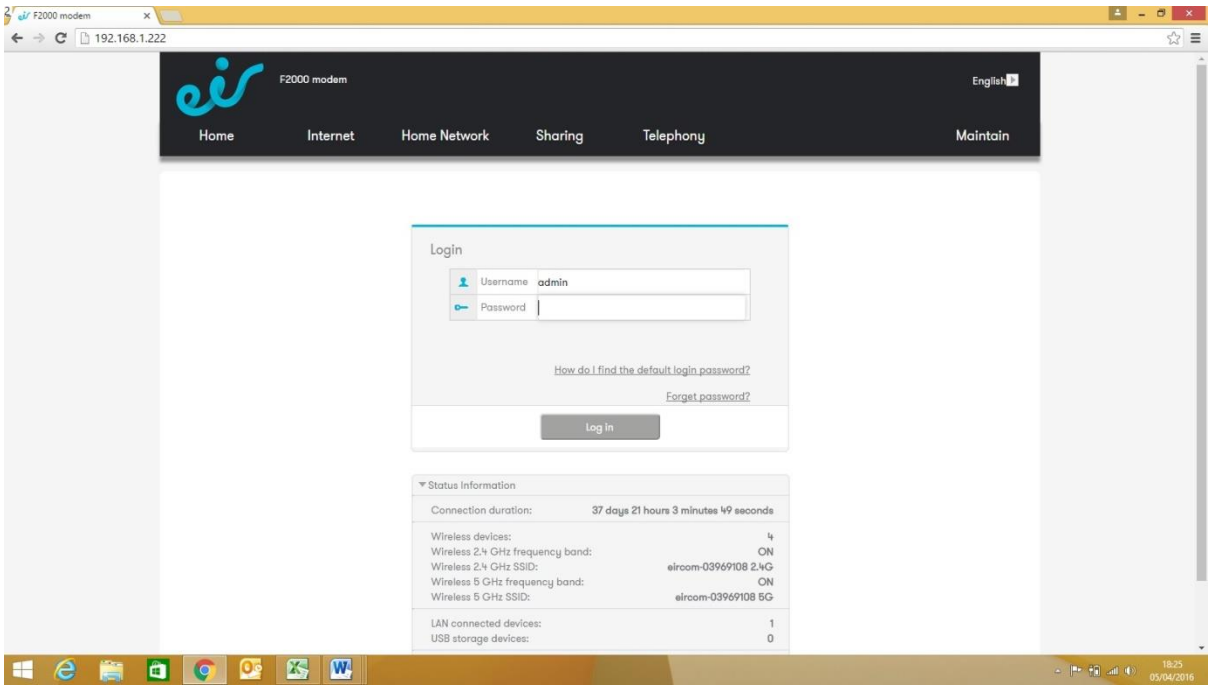
Mobile Device Networking & Management

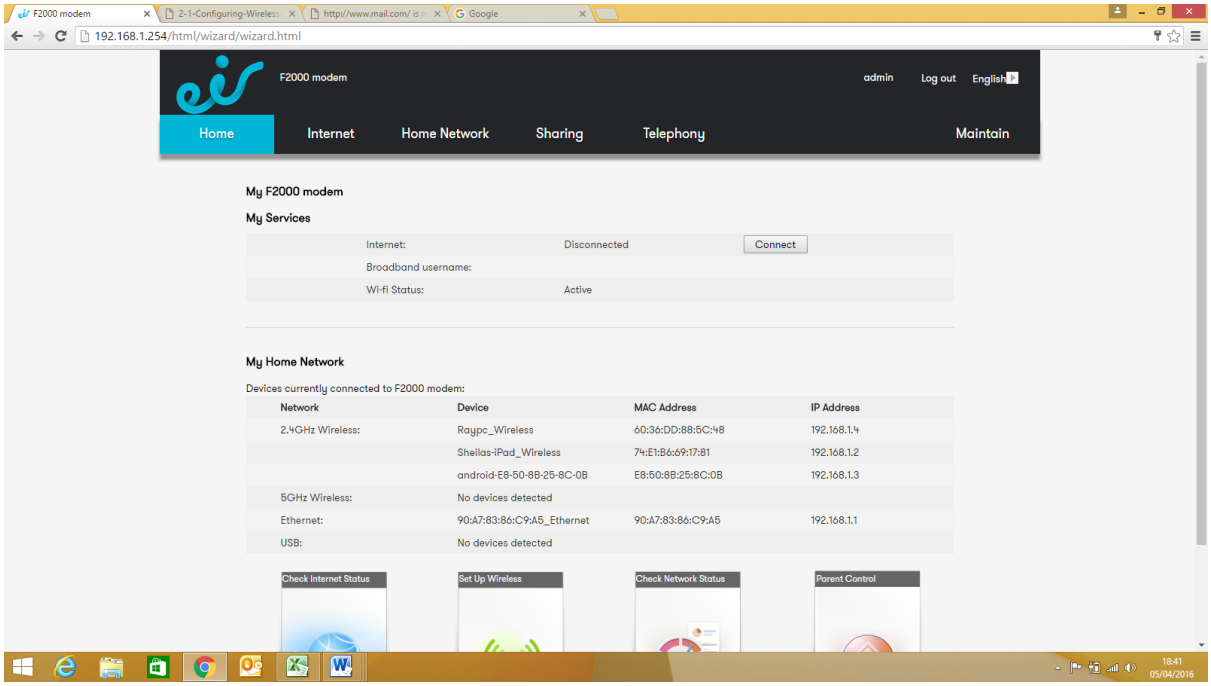
A practical exploration of mobile networking and device management that considers for example:

- configuring a wireless by router and adjusting common settings for example: DHCP, Encryption options, IP/MAC address filtering, port filtering and web content filtering
- connecting mobile devices to an appropriate data network
- installing applications on a mobile device
- updating applications on a mobile device
- considerations and procedures of performing system updates on a mobile device

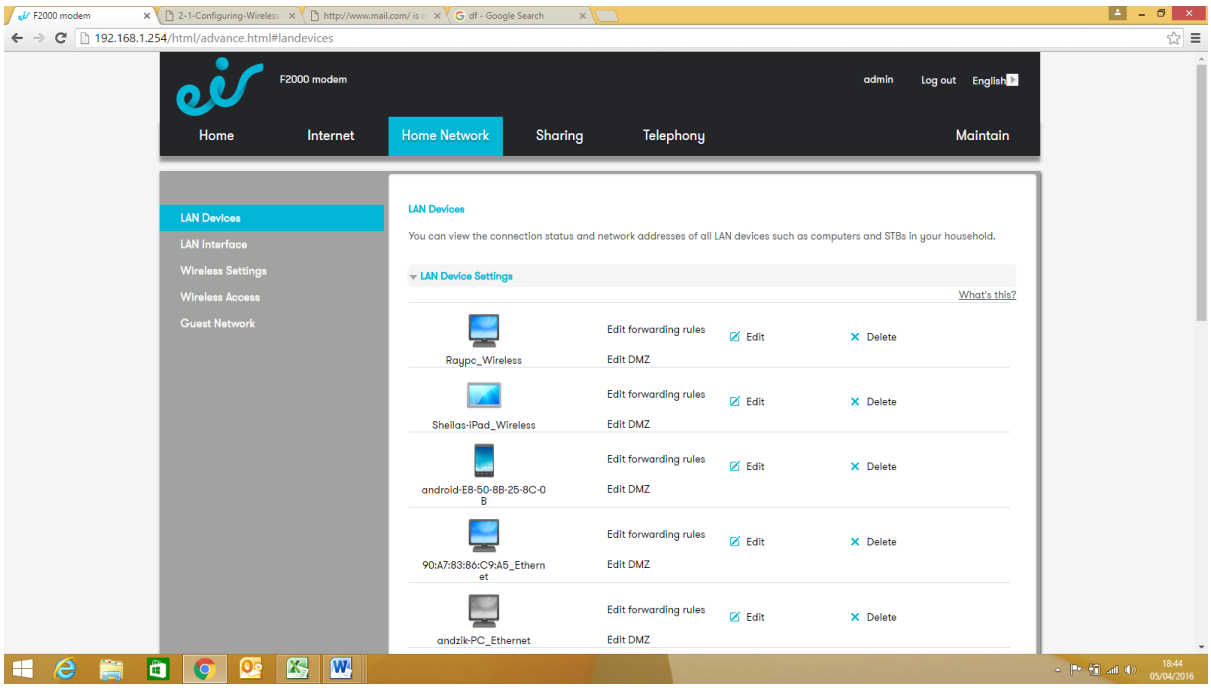
Eircom Modem/Routers



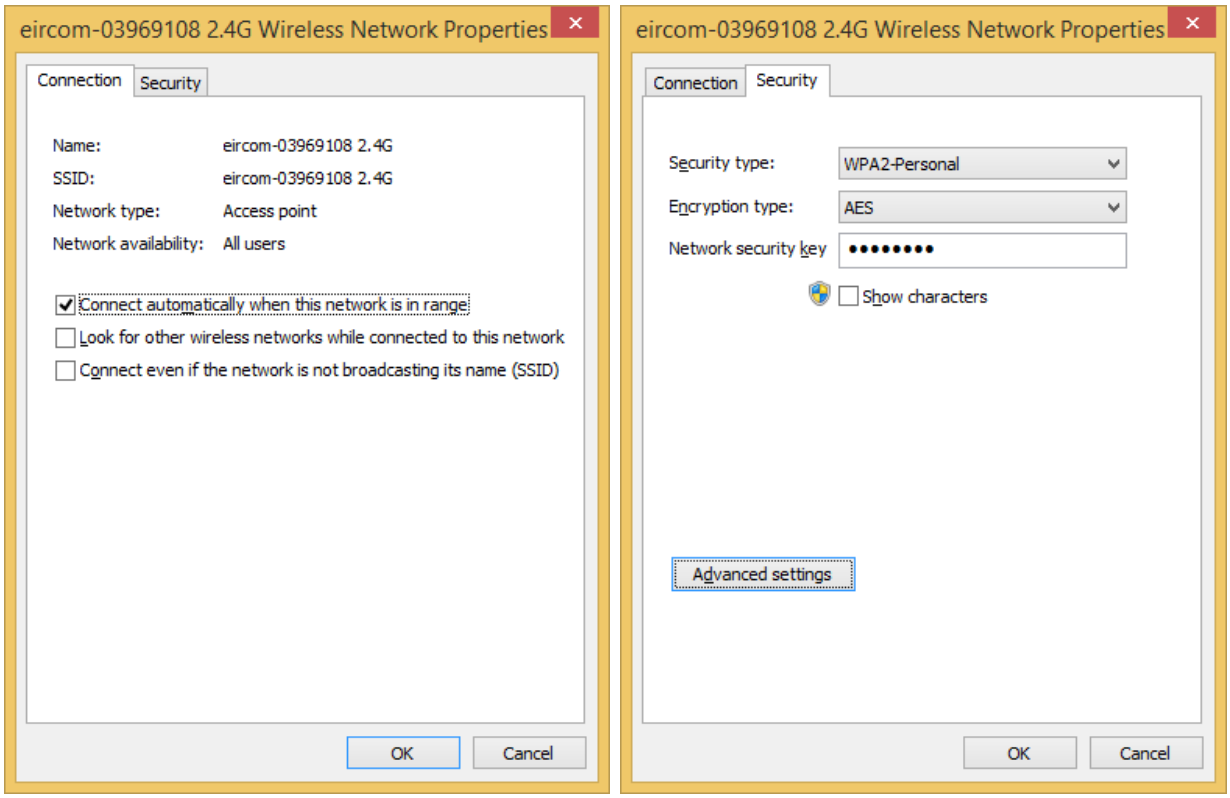
Once connected to your Wifi Router enter the IP address (eg 192.168.222) in the address bar of your browser. Type admin and password (these details are usually printed on the rear of your router)


F2000 modem
192.168.1.254/html/wizard/wizard.html
F2000 modem
admin
Log out
English
Home
Internet
Home Network
Sharing
Telephony
Maintain
My F2000 modem
My Services
Internet:
Disconnected
Connect
Broadband username:
Wi-fi Status:
Active
My Home Network
Devices currently connected to F2000 modem:
Network
Device
MAC Address
IP Address
2.4GHz Wireless:
Raypc_Wireless
60:36:DD:88:5C:48
192.168.1.4
Sheilas-iPad_Wireless
74:E1:B6:69:17:81
192.168.1.2
android-E8-50-8B-25-8C-0B
E8:50:8B:25:8C:0B
192.168.1.3
5GHz Wireless:
No devices detected
Ethernet:
90:A7:83:86:C9:A5_Ethernet
90:A7:83:86:C9:A5
192.168.1.1
USB:
No devices detected
Check Internet Status
Set Up Wireless
Check Network Status
Parent Control

LAN Devices


F2000 modem
192.168.1.254/html/advance.html#landevices
F2000 modem
admin
Log out
English
Home
Internet
Home Network
Sharing
Telephony
Maintain
LAN Devices
LAN Interface
Wireless Settings
Wireless Access
Guest Network
LAN Devices
You can view the connection status and network addresses of all LAN devices such as computers and STBs in your household.
LAN Device Settings
What's this?
Raypc_Wireless
Edit forwarding rules
Edit DMZ
Edit
Delete
Sheilas-iPad_Wireless
android-E8-50-8B-25-8C-0B
90:A7:83:86:C9:A5_Ethernet
andzik-PC_Ethernet

Security and Encryption



Difference between "encryption" and "security"

Coming back to the millions upon millions of leaked passwords: most modern web services are not foolish enough to store passwords in plain text, but there's another option that's just as bad, and it's the one thing that all these password leaks have in common.

But first, some background:

Basically, modern web services never store your password in the database. They put it in a cryptographic blender that frappés your password, takes a snapshot of the blended version, and stores that. When you type your password, it re-blends your password, and compares the new snapshot (or "hash") against the version in the database. Voilà, you've entered your password and it never gets stored anywhere.

But unfortunately, in 2012, part of Information Security 101 is knowing that "encrypted" does not necessarily mean "secure". All the leaked passwords from these major security breaches were stored with SHA–1 encryption. Technically, they were encrypted, but it was essentially the same as storing them in plain text.

SHA–1 encryption was invented at a time when the most powerful supercomputer couldn't crack it. But you're probably reading this on a machine that is 1000 times more powerful than the world's most powerful supercomputer when SHA–1 was created. Think of an unsalted SHA–1 hash as one of those decoder toys in your Frosted Flakes that contains a secret code that can only be revealed by a

transparent red plastic film. Suffice it to say, you do not want your passwords encrypted in an unsalted SHA–1 hash.

“Salting” the hash is like throwing an additional ingredient in the blender with your password. We hang on to the secret ingredient, and throw it in when we store or check your password. The snapshots still match, but it would be impossible for someone to un-blend your password and make any sense of it.

Lastly, like many modern Ruby on Rails applications, we use an encryption tool that is many, many times more secure than SHA–1 (called bcrypt). The encryption it’s built on stays difficult to attack, even as computers become more powerful. It would be essentially impossible for this algorithm to be decrypted, but if somehow it happened, the result would be useless due to the use of a salt.

Security *(Note the higher your security level often slows the user experience)*

If we could sum up our philosophy regarding security in three words, it’d be “secure by default”. Rather than a large, complex stack of unproven technologies, Bloomfire chooses a small set of tools that are battle-tested enough to prevent most common entry points, by default.

In the security world, there’s a bit of a tug-of-war between information security and user experience. Often, an idea for improving user experience directly impacts security, and a security feature directly impacts the experience for our customers.

Ultimately, our focus will always be centered on creating a great experience for our customers. And though we try to strike the right balance, we’ll default to the most secure option available, because we think knowing your information is secure is crucial to having a great experience with any web software.