



**NTNU – Trondheim**  
Norwegian University of  
Science and Technology

# Utilizing Mesh Potatoes in Emergency Situations

**Esther Bloemendaal & Ida Malene Hassel Øveråsen**

Submission date: May 2014  
Responsible professor: Norvald Stol, ITEM  
Supervisor: Sjur Eivind Usken, Aarbakke Innovation AS

Norwegian University of Science and Technology  
Department of Telematics



## Abstract



## Preface

This study was performed as a master's thesis on behalf of the Department of Telematics at the Norwegian University of Science and Technology. This report is the final result of this master's thesis and is worth 30 ECTS points. The study was conducted between January and June 2013. The project description was outlined in cooperation with our project supervisor Sjur Usken from Aarbakke Innovation AS.

We would like to thank Sjur Usken who has guided us throughout our project, and contributed with helpful ideas, feedback and support. We would also like to thank our project professor Norvald Stol for his helpful feedback, and insightful and motivating comments. We would like to thank Sigurd Albrektsen for helping us put together the box. We thank the Village Telco community that have answered our questions and helped us along the way. Finally, we would like to thank everyone who helped us proof-read our report.

Trondheim, June 11, 2014

Esther Bloemendaal

Ida Malene Hassel Øveråsen



# Contents

<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>xi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Problem Description . . . . .	2
1.3 Methodology . . . . .	3
1.4 Limitations . . . . .	3
<b>2 Background</b>	<b>5</b>
2.1 Village Telco . . . . .	5
2.2 Mesh Potato . . . . .	5
2.3 Relevant Technologies . . . . .	8
2.3.1 OpenWrt . . . . .	8
2.3.2 Telnet and SSH . . . . .	8
2.3.3 Mobile Ad Hoc Networks . . . . .	8
2.3.4 Wireless Mesh Networks . . . . .	9
2.3.5 Routing Protocols . . . . .	9
2.3.6 B.A.T.M.A.N . . . . .	10
2.4 The Cost Structure and Revenue Model(s) of Village Telco Today . .	12
2.5 Comparison of Village Telco and other Telecommunication Companies	12
<b>3 Refugees and IDPs</b>	<b>13</b>
3.1 Definitions . . . . .	14
3.2 Statistics . . . . .	14
3.3 Interview with CARE - Dadaab Refugee Camp . . . . .	15
3.4 Interview with Norwegian Refugee Council . . . . .	18
3.5 Life in Camps for Refugee Women . . . . .	19
<b>4 Main Work</b>	<b>21</b>
4.1 Set-up of the Mesh Potato . . . . .	21
4.1.1 Configuring the Mesh Potato . . . . .	22

4.1.2	Upgrading the Mesh Potato . . . . .	22
4.1.3	The SECN Web Interface . . . . .	24
4.2	The Emergency Box . . . . .	25
4.2.1	Previous/similar work . . . . .	26
4.2.2	Key Components . . . . .	26
4.2.3	Creating the Emergency Box . . . . .	26
4.2.4	Battery and Charging Calculations . . . . .	28
4.2.5	Possible Improvements . . . . .	30
4.3	The Process of Quick Roll-Out . . . . .	31
4.3.1	Script . . . . .	31
4.3.2	With the MP02 . . . . .	31
4.3.3	Get Started - How to Use the Box . . . . .	32
4.3.4	How Connect the MP to Cabled Internet . . . . .	32
4.3.5	How Connect the MP to Internet via PC . . . . .	33
4.3.6	How Connect the MP to Cellular Network . . . . .	36
4.3.7	How Connect the MP to Satellite . . . . .	36
4.4	Up-Link . . . . .	36
4.4.1	Internet via Telephone-line . . . . .	36
4.4.2	Cellular Network Technologies . . . . .	42
4.4.3	Satellite . . . . .	43
4.4.4	Summary Up-Links . . . . .	43
4.5	Future Internet Access Methods . . . . .	44
4.5.1	Google's Internet Balloons . . . . .	44
4.6	Apple's Mesh Network . . . . .	46
4.7	Different Scenarios Where a Quick Roll-out is Necessary . . . . .	47
4.7.1	Natural Disasters . . . . .	48
4.7.2	Temporary Refugee and IDP camps . . . . .	50
4.7.3	Festivals . . . . .	50
4.7.4	Breakdown of Mobile Towers . . . . .	50
<b>5</b>	<b>Discussion</b>	<b>51</b>
<b>6</b>	<b>Conclusion</b>	<b>53</b>
	<b>References</b>	<b>55</b>
	<b>Appendices</b>	
<b>A</b>	<b>Interview with Care</b>	<b>61</b>
<b>B</b>	<b>SECN-1.1 User Guide</b>	<b>65</b>
<b>C</b>	<b>SECN-2.0 User Guide</b>	<b>105</b>



# List of Figures

2.1	MP01 . . . . .	6
2.2	MP2 . . . . .	7
2.3	Cellular network vs. MANET . . . . .	9
2.4	Example of a Wireless Mesh Network . . . . .	10
2.5	Ad Hoc routing protocols . . . . .	11
2.6	Originator Message in B.A.T.M.A.N . . . . .	11
4.1	Flashing the Mesh Potato . . . . .	24
4.2	Web interface . . . . .	25
4.3	The composition of the emergency box . . . . .	28
4.4	Number of mobile-cellular subscriptions . . . . .	42
4.5	Project Loon: Balloon-powered internet for everyone. . . . .	44



# List of Tables

3.1	Refugee statistics - Comparing 2010 and 2012 [1, 2] . . . . .	15
4.1	The components of Emergency Box . . . . .	27
4.2	Advantages and disadvantages - Up-links [3]. . . . .	43



# Chapter 1

## Introduction

### 1.1 Motivation

Village Telco is an organization that aims to provide affordable communication in forms of data and voice services where no other companies can, or are willing to do so. Village Telco provides a “plug-and-play” solution with low cost voice and data service. While designed for the developing world, Village Telco’s solution can be applied anywhere where people wish to take control of their own telephone infrastructure.

This solution is delivered using an inexpensive fixed mesh WiFi delivery system called the Mesh Potato. The Mesh Potato unit is based on the open-source operating system, OpenWRT. Open Source telephony software combined with the latest wireless networking technology creates the potential for people to operate their own community phone systems. Mesh Potato networks have no dependence on existing telecom infrastructure, and can relatively easily be deployed anywhere in the world. It can either be deployed as a stand-alone solution or as an extension to existing technologies. Village Telco’s solution has been deployed in several countries around the world: from East-Timor and Nepal in Asia to several African and South America countries. The installed bases vary from 10 to several hundreds of Mesh Potatoes.

As of now the Mesh Potato has mainly been permanently deployed in small villages where the existing telecommunication systems are limited, non existent or too expensive. There are many scenarios where there is need for a solution that easily and fast can provide people with telephone communication and Internet, both within a community, and with the outside world. These scenarios span from natural disasters, post-conflict situations, temporary refugee camps and IDP (internally displaced person) camps, to the use at festivals, when a mobile tower is non-functioning, or during a blackout.

Our idea is to make an “emergency box” that consist of a Mesh Potato, a telephone, rechargeable battery, on/off switch and a solar panel to charge the battery. All this

## 2 1. INTRODUCTION

will be contained inside a robust and waterproof suitcase. All packed together and ready to go in any situation, at any time, anywhere in the world.

Our main focus is to provide the people with Internet access, it is crucial to have the possibility to communicate with the outside world during an emergency situation. In order to get Internet into the mesh network formed by the Mesh Potatoes, at least one of the Mesh Potatoes must be connected to an access network. Which type of access network that is available depends on the location. Some places there might exist stable landlines, other places not. Then an option could be to use satellite or cellular networks.

We hope to expand the potential of the Mesh Potato through our portable solution. We want to make it quick and easy to deploy, thus making it more useable in emergency situations. This does not only benefit the locals, but also makes the job easier for relief organizations. We want to provide communication where there are none, and believe that with the “emergency box” time would be spared and lives can be saved.

### 1.2 Problem Description

An area that has not been fully explored is the use of Mesh Potatoes in emergency situations, like natural disasters, post-conflict situations, etc. Another area to be considered is the use of Mesh Potatoes in refugee camps, where many people quickly gather in a new location. In both situations, the need for communication is essential. Key factors of usage are quick roll-out and usability. Easy to use communication is extremely important in crisis situations, both communication within the camp and outgoing communication with the rest of the world. It is important that all affected have easy access to helpful information, as this could mean the difference between life and death in some situations. In refugee camps with thousands of people, registering and reuniting people can be a difficult task to solve. Communication technology, like the Mesh Potato, could be revolutionary in situations like these.

We will look into how communication is handled by Norwegian emergency relief organizations today, what tools they are using, and if their way of communication could be improved with the Mesh Potato. In addition to this, we will look into other existing tools, and explore the possibilities to combine them with the Mesh Potato for a better product.

\*Skrive om hva vi endra til, og hvorfor vi endra.

### Research questions

- How can the Mesh Potato be used in different an emergency situations?
- How can an emergency box be developed? What components, set-ups and configurations are necessary?
- Look at different situations where there is a need for a go-box. What are the different needs in the different situations?
- Look at the process of quick roll out. What measures can we do in advance to make it as easy and fast as possible to connect the go-box to and internet connection.

### **1.3 Methodology**

Research

### **1.4 Limitations**

Our main limitation was the amount of time we had available to finish our masters thesis, we only had 21 weeks at our disposal. When entering a new field it takes some time to understand the technology used. The hardware and technology we have been working with is mainly used in rural areas. Another limitation is money, we tried to get funding from Engineers without Borders, but unfortunately they had no funding available.





# Chapter 2

## Background

### 2.1 Village Telco

### 2.2 Mesh Potato

The Village Telco concept was developed in June 2008 during a workshop at the Shuttleworth Foundation in Cape Town, South Africa. The main goal was to develop an inexpensive system to provide rural and under-served areas with affordable telephone communication [4]. The workshop included participants like open hardware pioneer Dawid Rowe, and Elektra, the developer of B.A.T.M.A.N (for more information about B.A.T.M.A.N see section 2.3.6) [5]. The purpose of the workshop was to develop a business model, as well as a prototype for a Village Telco. Initially the idea was to use low cost VoIP headsets. At that time it was the most viable and convenient way to deliver telephone services to the customers. The wireless VoIP telephones have small antennas, which became a problem. The nodes could not be more than 100 meters away from each other in order to have a reliable connection. This required more nodes in order to cover a desirable area. This factor drastically increased the start-up costs for a village. In order to keep the cost down, it was also important to keep the number of access points (APs) down. A mesh network has a larger range, and one suggestion was to use a small mesh device like an Open Mesh AP and connect a SIP phone to it. This solution would solve a lot of the problems regarding range, antenna and number of access points, but the idea was still an expensive option. The challenge was to create something that would be simple enough to be configured, and scaled by local entrepreneurs with limited technical skills. In addition to this it was important to keep the cost down. The two key cost factors that emerged in the scale-up of a Village Telco were the cost of the customer's phone and the power supply. It was clear that the power supply was the most important factor, and that they had to look at other, and cheaper options regarding the customers phones [5]. During the debating, Rael Lissoos took an Analogue Telephone Adapter (ATA) and an Open Mesh AP, held them together and said; *"we need these two devices in*

*one*". This point was the birth of the Mesh Potato, fully based on customized open hardware and software design. The name "Mesh Potato" comes from combining the words mesh, POTS (Plain Old Telephone) and ATA. "Patata" is the Spanish word for potato, and hence the name Mesh Potato. The Mesh Potato is a mesh enabled Wi-Fi device, with the possibility to connect any inexpensive regular phone and IP device. [6]

The first generation of the Mesh Potato is shown in Figure 2.1. This device was designed to be used in rural areas. It can be deployed and run anywhere in the world, relying only on a low, but stable, power supply. The Ethernet port, the Foreign eXchange Station (FXS) ports, and the power port are robust and designed in order to handle all weather conditions, poor power conditions, lightening and static electricity. In addition to this, the Mesh Potato comes in a waterproof box for outdoor mounting [7].

The Mesh Potato combines the features of a 802.11bg Wi-Fi router with an Analogue Telephone Adaptor (ATA) [8]. The ATA converts the signal from a standard telephone, into the digital signal needed to connect to the Internet and use the SIP protocol [4]. The device is based on the Atheros chipset that is used by OpenMesh, and runs OpenWrt (see section 2.3.1 for more information) and B.A.T.M.A.N. (see section 2.3.6 for more information). Each Mesh Potato provides a single fixed telephone line to the end user. The MPs are connected together via a mesh Wi-Fi network, and configure themselves automatically to form a peer-to-peer network, greatly extending the range of the network over regular Wi-Fi. This enables the phone calls to be made independent of landlines and telephone towers, and creates the basis for the "plug-and-play" solution.

As mentioned, the Mesh Potato is based on open hardware, as well as open



**Figure 2.1:** The first generation Mesh Potato, MP01.

software design. Everything is kept open in order for any third party to test, set standards, and give feedback. Key goals during the development was to minimize the binary blobs (a closed source binary-only driver that has no publicly available source code [9]), minimize closed software and make the hardware open.

The mesh network can be connected via a backbone link to the rest of the world by using VoIP gateways. No cell phone towers, no land lines, and no telecommunication companies are required. A Village Telco is a community owned telephone service, allowing a local entrepreneur to roll out the Village Telco system only needing a server and the wanted amount of Mesh Potatoes. The mesh network is self-healing and self-organizing, meaning if one node goes down, B.A.T.M.A.N. routes the calls through other available nodes in the network [10]. In order to provide Internet to the mesh network, one of the Mesh Potatoes must be provided with Internet access. The internet signal enters the server in the Village Telco, this could for example be an existing internet café, with a broadband, link or satellite connection. The signal is transmitted to the super node. The super node consists of three external access points, and is placed high over ground, giving 360 degree coverage, with approximately 1 km range. The internet signal is then carried through the network from one Mesh Potato to another.

### Mesh Potato 2.0

The first generation of the Mesh Potato has sold over 2500 copies, and is deployed all over the world. In order to keep up with time, the constant technical development and the demand from the users, a new version of the Mesh Potato was introduced. The second generation became available to users August 2013. This device comes in a smaller box, as shown in Figure 2.2, and is sold to half the price of the first generation. One of the biggest differences is that the second generation has two Ethernet ports and is built on a new, and faster, chipset. It is also operating on new firmware.



**Figure 2.2:** The second generation Mesh Potato, MP2.

## Difference between MP01 and MP02?

### 2.3 Relevant Technologies

In this section we will go through some of the most relevant technologies used to develop and run the Mesh Potatoes. In order to understand how the Mesh Potato works, it is important to have a certain knowledge about the underlying technology.

#### 2.3.1 OpenWrt

OpenWrt is an embedded open-source operating system for routers distributed by Linux [11]. It is extensible and can easily be modified to suit any application, since it offers a file system with a package manager. OpenWrt provides (1) Free and open-source, (2) Easy and free access, and are (3) Community Driven [11]. This means that the source code is free and available to everyone, and that everyone has the opportunity to contribute to it.

#### 2.3.2 Telnet and SSH

Telnet is a TCP/IP protocol that enables the opportunity to remotely connect to a computer/device. In order to do this, telnet client software is necessary. The client becomes a virtual terminal, and through command line prompt one can remotely work with files and data [12].

Secure SHell (SSH) offers the same services as telnet, but is a more secure alternative. With SSH all data sent to and from the server is encrypted [13].

#### 2.3.3 Mobile Ad Hoc Networks

Mobile ad hoc networks (MANETs) are networks that do not rely on an underlying and fixed infrastructure (access points and routers), in other words "infrastructure-less". MANETs acts in a shared wireless media [15]. The structure of these networks change dynamically. Key factors describing MANETs is self-configuration, self-organization, self-discovery, and self-healing [16]. The members of the network are mobile and free to join, or leave, the network at any time [14]. MANETs are based on multi-hop forwarding. Each node acts not only as a host, but also as a router. The nodes themselves establish and maintain routes, and forward packets to other nodes if necessary. This enables communication between nodes that are originally not within each other's range [14]. MANETs are suited for use in situations where there are no fixed underlying infrastructure. A MANET can operate as a stand-alone solution, but can also be attached to the Internet.

### 2.3.4 Wireless Mesh Networks

A wireless mesh network (WMN) is a type of MANET [16]. The objective of a WMN is to serve a larger number of users with high bandwidth access. As mentioned before, MANETs are "infrastructure-less" and they have self-configuration, self-organizing, self-healing and self-discovering features. WMNs share all these characteristics, except from the infrastructure part. WMNs are often a collection of routers called mesh routers (MRs). These MRs are usually stationary. The MRs can be employed for different use. One MR could for example be connected via cable to Internet, and then become an Internet gateway. Then this MR can provide Internet connectivity to the other MRs in the mesh network. A wireless mesh network consists of two parts; the backbone of the mesh (the MRs) and the clients of the mesh [16]. An example of a WMN architecture is shown in Figure 2.4.

### 2.3.5 Routing Protocols

Ad hoc networks and mesh networks creates several challenges when it comes to routing protocols. The routing protocols must be able to adapt quickly due to the topology changes. Figure 2.5 shows the different groups of the ad hoc protocols that exist. It is important that a routing protocol do not cause excessive overhead (extensive use of computer resources). Under the category flat routing, there are two types of routing protocols; proactive and reactive. *Proactive routing protocols* (e.g. OLSR) are table driven [17]. Every network node has a routing table for forwarding of data. To obtain stability, each node broadcasts and modifies the routing table periodically. Proactive routing protocols are suitable when there are few nodes in the network. The routing table is periodically updated, hence the overhead exceeds the desired value when there are a high number of nodes in the network. In contrary



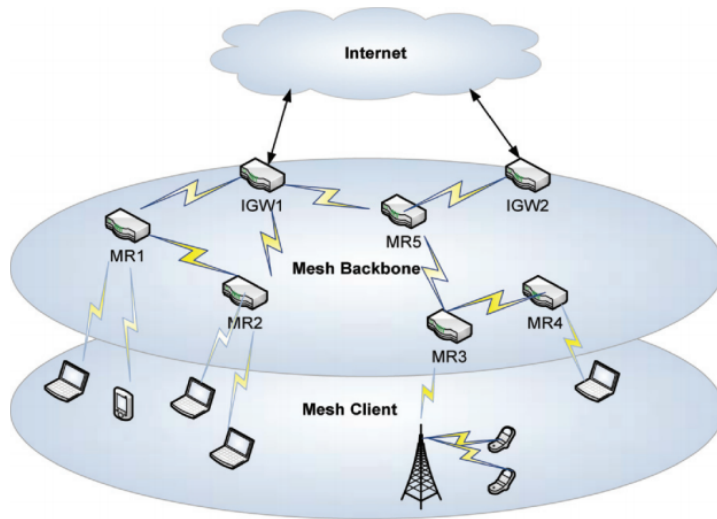
**Figure 2.3: Cellular network vs. MANET.** This figure illustrates the difference between a regular cellular network and a mobile ad hoc network [14].

to the proactive routing protocols, *reactive routing protocols* (e.g. AODV) are on demand. Since they are on demand, the overhead is significantly lower. These protocols utilize flooding. The network is flooded with the route request (RREQ) in order to set up the route. The reactive routing protocols do not have a up-to-date routing table like proactive routing protocols [17]. Routes are only set up to nodes they communicate with, and these routes are only kept alive while they are needed [14]. As shown in Figure 2.5, there are several different protocols under proactive and reactive.

### 2.3.6 B.A.T.M.A.N

Better Approach To Mobile Adhoc Networking (B.A.T.M.A.N) is the routing protocol utilized in the networks formed by the Mesh Potatoes. B.A.T.M.A.N is a proactive routing protocol for wireless ad hoc networks. This includes MANETs [18]. This protocol was developed as an alternative to OLSR (Optimized Link State Routing) [19]. Like mentioned before, routing protocols must be able to adapt quickly to topology changes. B.A.T.M.A.N was made to be a more efficient routing protocol in this area, since it employs a new method for discovering routes. The nodes in the network broadcasts a OGM periodically, like shown in Figure 2.6. A OGM is a Originator Message which contains:

- The address of the node



**Figure 2.4: Example of a Wireless Mesh Network.** This figure illustrates the architecture of a typical WMN [16].



**Figure 2.5:** Different groups of ad hoc routing protocols [15].

- Sequence number
- TTL (Time to live)

The address and the sequence number enables identification of a packet and duplicate detection.

Information about the nodes that are accessible via single-hop or multi-hop are maintained and updated [18]. Every node updates its routing table each time it receives an OGM. The routing table includes information about [19]:

- **Originator Address:** This is the source address of the node that sent the OGM.
- **Current Sequence Number:** The sequence number of the last OGM. This is used to discover if there are any duplicates or any information that is outdated.
- **Sliding Window:** A list of sequence numbers that is stored for each originator and each previous hop, i.e. for the neighbour node that forwarded or originated the OGM, as shown in Figure 2.6. This is used to decide which next hop is best for each destination.



**Figure 2.6:** Originator Message used in B.A.T.M.A.N [19].

When a node receives an OGM it will decrease the TTL, and then forward it to the neighbour nodes. The same OGM can arrive to a node, but from different paths. In this case, only the first copy is preserved.

### **Simple Unified Dashboard for mesh networks**

Simple Unified Dashboard (SPUD) for mesh networks is a tool for visualization made for B.A.T.M.A.N mesh networks, and for the users of the networks [20]. The Simple Unified Dashboard is, like the name indicates, a dashboard based on PHP which is designed to be simple. It communicates with the B.A.T.M.A.N visualization server. The dashboard makes it possible to monitor the link status of the networks, by displaying real time wireless link status. Other features are client management and customization. The software is written in CakePHP and for visualization SPUD uses Google Maps API 1.3 [20].

## **2.4 The Cost Structure and Revenue Model(s) of Village Telco Today**

## **2.5 Comparison of Village Telco and other Telecommunication Companies**



## Chapter 3

# Refugees and IDPs

Our initial approach for the report was to look into refugee camps, with the focus on Norwegian relief organizations. The aim was to get an understanding of how refugee camps are run and how the communication is both within the camp and to the outside world. We wanted to look into how the Mesh Potatoes could be utilized to improve the communication. We conducted some research trying to get an overview and general understanding. In addition we carried out interviews with people from Norwegian organizations namely Care and the Norwegian Refugee Council. This gave us more information and understanding on the area, but also showed us that refugee camps is an extremely large area, and the differences in the unique camps are huge depending on country and government law, size of the camp, lifetime of the camp, etc. In some countries the Internet and GSM infrastructure is well established, and people have smartphones and laptops. While in other countries this is far from the case, and their only way of receiving news is by radio or mostly by word of mouth. Some camps are under strong technological restriction because of country law.

The differences are so big it is hard to find general information, simply because there are no general information. No camp are the same, no situation can be compared to each other. Which made it hard to decide in what direction to focus. A research, as the one first intended, would require a close collaboration with a relief organization to give full focus both to us and the research we are conducting on behalf of Village Telco. It was also difficult to create a stable connection and collaboration with one of the Norwegian help organizations. A collaboration like this would require a lot more time than what we have at disposal.

During our research and work in the first months we could see our report going in a different direction. The area became too big to get a grip on. We decided to direct our focus onto natural disasters, and how an emergency box can be used to quickly get internet access and in that way coordinate help, more on this in chapter 4. This chapter contains the research we have done on refugee camps. We will go through some general statistics to get an idea of the life in the camps and the development

the over the last years, and summaries of the interviews with Care and Norwegian Refugee Council.

We therefore chose to co-operate with a smaller organization called Making Change. Making Change is a non profit relief organization where military veterans can use their experience to make a change somewhere in the world.

### 3.1 Definitions

It is fairly common to think of every person that is displaced as a refugee, but this is not the case. It is important to separate between a refugee and an internally displaced person.

**Refugee.** The definition of a refugee is a person who have been forced to leave their home country because of war, violence or persecution. A refugee often has a justifiable fear of persecution for reasons as religion, political opinion, race, nationality or membership in a certain social group. For these reasons they are not able to, or afraid to return to their home country. The leading reasons for refugees fleeing their home country is war and ethnic, religious and tribal violence [21].

**Internally Displaced Person.** An internally displaced person (IDP) is a person that has been forced to leave their home and village for some reason and are a refugee in its own country. The main distinction between an IDP and a refugee is that the person has not crossed any country borders. Unlike refugees, the IDPs are not protected by any international laws nor are able to receive many types of aid. In the last years the number of IDPs have drastically increased, mostly due to the conflicts between countries [21].

**A stateless person** is someone who does not have citizenship in any country. A citizenship is a legal bond between an individual and the government in that country.

### 3.2 Statistics

We have looked at some of the refugee statistics presented by UNHCR from 2010 and 2012. The following section will enlighten some of their findings.

In 2010 the majority of the refugees came from Afghanistan, Iraq and Somalia [1]. In both 2010 and 2012 Pakistan was the country which hosted the most refugees. By the end of 2012 45.2 million people were displaced by force. According to UNHCR this is the largest number in 20 years. The report for 2012 show that 55 % of the registered refugees came from countries affected by war, e.g Syria, Afghanistan, Sudan, Iraq and Somalia. The crisis in Syria has been a major factor to displacement,

the whole of 647,000 people have be forced out of the country [2]. Table 3.1 shows the drastic increase of Syrian refugees from 2010 to 2012.

An observation made is that the number of Somali refugees has increased for 770,154 refugees in 2010 to 1,136,100 refugees in 2012. There has been an ongoing conflict in Somalia ever since the Siade Barre regime collapsed in 1991. Since 1991, many Somalis have been displaced. The number of displaced persons have gone in waves. In 2011-2012 there was a famine in Somalia, and this caused not only many deaths, but the displacement of many humans [22]. This is one of the reasons for the increase of Somali refugees from 2010 to 2012.

In contrary to the increase of Somali refugees between 2010 and 2012, the number of Iraqi refugees had decreased from 1,683,579 refugees in 2010 to 746,400 refugees in 2012. Before the Syrian civil war started in 2011, there were many Iraqi refugees who had fled to Syria due to the invasion led by the U.S. When the Syrian civil war began, the situation reversed. Many Syrian people sought shelter in Iraq, and many of the Iraqi refugees returned to their homeland. This is one reason for the drastic decrease of Iraqi refugees from 2010 to 2012. Although many Iraqi refugees went back to Iraq, they remain displaced. This situation have brought the number of Iraqi IDPs up to approximately 2.8 million [23].

**Table 3.1:** Refugee statistics - Comparing 2010 and 2012 [1, 2]

Information	2010	2012
Number of people forcibly displaced worldwide	43.7 million	45.2 million
Number of refugees from Afghanistan	3,054,709	2,585,600
Number of refugees from Somalia	770,154	1,136,100
Number of refugees from Iraq	1,683,579	746,400
Number of refugees from Syria	18,452	728,500
Number of refugees hosted by Pakistan	1,900,621	1,638,500
Percentage of refugees that are female	47%	48%
Percentage of refugees that are children (below 18)	47%	46%
Number of individual asylum applications lodged by unaccompanied or separated children	15,500	21,300

### 3.3 Interview with CARE - Dadaab Refugee Camp

We got in contact with Mary Muia from CARE. She is a program assistant at CARE International in Dadaab, Kenya. We sent her a questionnaire with questions

about Dadaab refugee camp, with focus on means of communication. The following paragraphs contains information both from different articles referred to in the text, and from the answers from the questionnaire. See Appendix A for the full questionnaire. Dadaab is the largest refugee camp in the world, and is located in Daadaab, Kenya [24]. It was created in 1991 [25]. Dadaab was created by the government of Kenya and UNHCR to host Somali refugees displaced by civil war. Over the years, the camps have also hosted other nationalities, from the Horn of Africa, the Great Lakes and East African regions. These people constitute less than two percent of the camp population. In April, 2013, there were 423,496 registered refugees in the Dadaab camps. 51 % of these were female and 58 % were younger than 18 years old. Also in 2013, UNHCR and its partners decided to conduct a verification exercise to ascertain the current population. The reason for this was that many of those who had arrived in 2011 due to the famine had returned home. As of February, 2014, the current population stands at 369,294. The lead agency for this camp is the UN High Commission for Refugees (UNHCR) [24]. In addition to UNHCR, major international humanitarian agencies like Care, Save the Children and the International Rescue Committee are active helpers in the Dadaab refugee camp. These agencies provide the refugees with critical services (e.g. food, housing, sanitation and medical help). This is an extremely challenging task in refugee camps, especially when they reach this size. During the recent years, the terror group Al Shabaad (Somali-based) have intensified their misdeeds in and around the Dadaab refugee camps. This has made the situation even tougher for the refugees and the relief agencies. Muia stated that the biggest challenges in the camps are lack of enough space to accommodate everyone, and lack of enough funds to take care of all the needs of the refugees. Another challenge is the language barrier between the humanitarian staff and the refugees. Many of the staff members neither speak nor understand the Somali language, and as many as 95.6% of the refugees are Somali.

Muia explains how the registration process is handled; When a new refugee enters the camp, the refugee reports to a UNHCR reception desk. There the refugee is given a temporary registration, while pending full registration. Upon arrival, the refugees are given information about available services, and which agency is handling what service. Immunizations, medical attention, emergency food supply, tarpaulins, sleeping mats, jerrycans for fetching water and kitchen sets are issued to new arrivals. This is to help them start their new lives in the camp.

To improve the situation in Dadaab, communication is crucial. In 2011, a group consisting of people from NetHope, Inveneo and the USAID Global Broadband and Innovations Program gathered to discuss ways to improve the means of communication in Dadaab [24]. NetHope is a consortium of over 30 international Non-Governmental Organizations (NGOs) [26]. NetHope works with improving connectivity, with the help of information technology, among relief agencies. The aim of this project, called

DadaabConnect, was to bring forward more reliable Internet, and find ways for agencies to communicate better internally [24]. The group put together teams that travelled to Kenya to investigate the conditions in the refugee camps, and to find out what they could implement. It was clear from the feedback they got that a better communication system was needed, and that it would make the humanitarian work much easier. It would improve the coordination and the security in the camps. Improvements of these aspects gives the humanitarian agencies better working conditions, and makes it easier for them to help the refugees with critical services. Inveneo started working with Cisco's Tactical Operations (TacOps) to install and configure a local high-speed network [27]. They also entered a partnership with a local Kenyan mobile and landline telecommunications service provider called Orange. The reason for this was that they wanted to extend the Dadaab compound with new data services. This could be done by using Inveneo's long-distance Wi-Fi solutions. The data services that were added included services requested from the Dadaab aid community. "DadaabNet", a high-speed network, was created in cooperation between Inveneo and TacOps. This network connected the NGOs locally, and made it possible for the agencies to easier communicate internally (VoIP telephony, file sharing etc.). Following this, in March 2012, they started the training of technicians. These technicians were people from Orange, from the technical staff of the NGOs and from Inveneo's staff. The training took place both in classrooms and in the field, in order to give the technicians a wide understanding. The results from DadaabConnect has been great. The humanitarian agencies has gotten better working conditions, due to the improvements in means of communication. Other positive outcomes is that the network is more reliable and cost effective.

Muia did not specifically mention this project in her answers, but answers on our questions about means of communication within the camp, and with the outside world. She states that CARE as an organization has invested in communication systems in cooperation with ISPs in the capital city of Kenya, Nairobi. Through this cooperation the camp staff are assured to get access to Internet for both official and social use. Several Kenyan telecommunication companies have put up equipment in the camp area, and the camps are therefore provided with access to mobile communication and Internet. Although this is set up, Internet and telephone service outages are fairly common. In addition to mobile communication and Internet, there are radio station services and access to digital television. CARE use telephone services to reach out to refugee staff. 50% of the refugees have access to mobile phone services. Posters and radio are also used to reach out. Word by mouth (e.g. over speakers) is also a communication technique employed. There are two main telecommunication providers in Dadaad, hence little competition. This makes the prices higher. We asked Muia how the refugees can afford having their own mobile phone, when the costs are high. She says that many refugees have been in the camps for a long time, and therefore have had the time to establish small businesses which gives them some

profit. While others get money sent from their relatives.

### 3.4 Interview with Norwegian Refugee Council

We had a Skype interview March 12, 2014, with Katrine Wold from the Norwegian Refugee Council (NRC). The aim of the interview was to hear a little bit about her work in refugee camps and how the situation in the refugee camps are today, with main focus on means of communication. Katrine Wold has been working for NRC for many years, and also has a background from United Nations (UN). She has worked in emergency and crisis situations abroad. She is specialized in camp management and coordination. In recent years she has been responsible for education, and have had the main focus on youth. We asked her which refugee camps NRC is working in, but she could not give us a clear answer on that question. The reason for this is that NRC works in over 24 countries, and have, as of 2013, reached out to 4.4 million people. She makes it clear that there is a difference between internally displaced persons (IDPs) and refugees. An official refugee must cross a boarder, or else you are internally displaced. NRC works both with refugees and IDPs, and also with people who are affected by having refugees in their local area. NRC does not only help with operational issues in the camps, but they mainly offer services the refugees need. When dealing with refugees there exists international laws and regulations. These also states what kind of human rights exists. Everyone have rights! The vast majority of countries have ratified the UN refugee commission, which has been formed by the international society, UN, and authorities via UN's forums. The commission is an important premise when working with refugees. It is important to know which rights you have as a humanitarian worker, and which rights the refugees have.

We ask her about how communication within the camp is conducted. She takes Kenya as an example. NRC has been working in the largest refugee camp in the world, Dadaab Kenya, for many years. Some have the main responsibility for what is going on in the camp, and that is the authorities. They often ask the international community (e.g. NRC) for help. Wold states that it is then important to establish a good communication and information flow between the ones working in the camp (the different organizations). This communication takes places by either establishing coordination meetings and by other types of mechanisms. These meetings includes the relief organizations working in the camp, and the authorities. The goal is not to make a permanent home for the refugees, but that it is safe when they are in the camps and that they move on (either go home or find another place to live). Living in camps is a temporary life situation. She states the different types of communication; internally between the workers in the camp and communication with the refugees. It is important to establish open transparent coordination mechanisms, in other words ensure good forums where the refugees can communicate and inform the workers in the camp what their needs are. This can only be achieved by recognizing that

refugees is not a large mass, but individuals with different needs and different life situations. The humanitarian and authorities try to establish some sort of local elections. This means that the refugees can choose representatives who's job is to be in communication with the primary humanitarian managers in the camp. The reason for this is that it is impossible for the humanitarians to talk to 500 000 people. The communication between the representatives and the managers must be done either through meetings, or in an informal manner. Overall, this creates a communication pattern in the refugee camps. Wold states that there are a few places without mobile coverage, and that the majority of the refugees have a mobile phone. Mobile phones are used frequently in terms of distribution. Mobile phones are often used as a tool when goods (access to money, food etc.) gets distributed to the refugees. They can "add credit" to their card, and use this as "payment". This is an up-and-coming way of doing distribution. Mobile phones are also used to collect information, for example by sending the refugees surveys on their mobile phone.

In general, Wold states that methods of communication can be via mouth, radio, billboards, data communication, but this all depends on which camp and what is allowed in the camp. The law in the refugee camps depends on the national authorities. In some camps it is allowed to establish a data communication center, but in other camps this is illegal. It is important that when the refugees arrive to a camp that they get informed of the current situation, and what rights they have. The distribution of this information takes place primarily by someone called the camp management agency. They have the daily coordination responsibility for what is going to take place in the camp. It must be made clear to the refugees where they can obtain different types of services, and also what is expected of the refugees. It is important that the refugees at an early stage get the opportunity to contribute positively in the camp, or else they can end up with something called "dependency syndrome" (they feel incompetent and get totally dependent on external assistance).

Another question we asked her is how the refugees get registered in the camps. Here she states the importance of distinguishing between official and unofficial camps. The definition of a camp is that people are gathered together and live there. Registration is done in official camps, and then there are someone who is responsible for the operation of the camp. When refugees are registered they get an ID card. This ID card is very valuable, because it indicates that you, as a refugee, have access to the goods that are available in the camp. The registration procedures can vary, but most often there exists computer systems for the registration.

### 3.5 Life in Camps for Refugee Women

In this section we will shortly present some answers found in research done by Mari Maasilta, a Swedish post doctoral researcher, most relevant to our assignment. She

have looked at the use of oral and mediated communication by women living in refugee camps in Eastern-Africa.

Women are in general in a more vulnerable position when living in a camp, especially if they are single mothers. They may be solely responsible for taking care of the children in addition to sick and elderly family members, maintaining the household, preparing food, acquire water, and securing firewood. Collecting firewood for cooking is a necessity, but it forces women to walk far away, hence making them vulnerable for sexual assault. They have to turn to prostitution and other unhealthy and dangerous means in order to survive [28].

The means of communication vary greatly in the different camps. Some have internet connection and satellite TV, other barely have access to a radio. Even though radios are the most common media for communication, it is not given that all citizens in a camp have access to one. Often people would gather around the few radios that exists in a camp. One issue that limits the usage is the batteries. They are very expensive and hard to acquire. The women were interested in news regarding their place of origin. The use of cell phones are increasing. Even though the prizes are extremely high it does not stop people from calling relatives in Europe and other places in the world [28].

Information walls and word of mouth is often used in order to spread practical information within the camp and about camp activities. Word of mouth is also used in order to retrieve information about the world outside the camp. People visiting the camp were used as sources for information. Earlier studies have shown that social connections with neighbours works as an important medium to transport information, resources and services between individuals. These kind of networking have been used to find lost family members in big camps, as well as get financial help from abroad [28].



# Chapter 4

## Main Work

The main purpose of our study was to create an easy way to utilize the Mesh Potatoes in emergency situations and other situations where there are need for cheap and instant communication. Our main focus has been on providing Internet access to the mesh network, and the aspect of quick roll-out of the network. Internet access may be a vital way of communication in emergency situations, or just convenient in other situations. The process of setting up a mesh network providing Internet should be done quickly, since time is a crucial factor to consider in emergency situations. In this chapter we will describe parts of the set-up techniques for the Mesh Potatoes. Further on we will present our Emergency Box, which is a box including a Mesh Potato, telephone, solar panel, battery and a charging regulator. A box ready to go in any situations. We will describe how we made this box, and similar work that have been conducted previously on the area. When it comes to the quick roll-out, specific approaches for making a best practice for quick roll-out will be presented. We will present the different types of up-links available to provide the mesh network with Internet access, along with a manual on how to connect the network to the different up-links. At last we will describe some situations where the need for a quick roll-out of a communication system may be needed.

A big part of our work is gathering of information, and make this information more understandable for the common user. The existing user guides tend to be advanced and not adapted to people with little background knowledge on the area. Therefore we found it necessary to make the descriptions easier and more user friendly. The set-up procedures we explain in this chapter can also be found elsewhere.

### 4.1 Set-up of the Mesh Potato

The set-up process of the Mesh Potato consists of, among other activities, installing firmware, allocating IP-addresses and providing Internet to the network. The firmware used for the Mesh Potatoes is called Small Enterprise/Campus Network (SECN)

Firmware [29].

#### 4.1.1 Configuring the Mesh Potato

Configuring a Mesh Potato is the process of allocating an unique IP-address to the MP. Each of the MPs were assigned a static IP address, these addresses were not part of the LAN address space. The IP addresses are allocated in a predefined default address space 10.130.1.20. In order to change this IP address, the MP has to be connected via an Ethernet cable to a PC running Linux. The PC must be on the same subnet as the MP, in order to establish contact. When the PC is on the same subnet, the MPs web interface can be accessed via a browser. In the web interface, the IP of the MP can easily be changed. This description works for both versions of the MP. One difference is that you can change the IP on MP01 by using interactive voice response (IVR) commands (see page 28 in Appendix B). A second version of the MP02 is in development. This version has a telephone jack port (FXS daughterboard), which allows for IVR.

1. Set the PC to have the same subnet, by writing the following command in the terminal:

```
$ ifconfig eth0 10.130.1.120 netmask 255.255.255.0
```

2. Open a browser and type in "10.130.1.20". The web interface will then appear.
3. In the web interface, under network, change the IP address field to "192.168.1.x", where x is the unique number for the specific MP. This number should be between 21-99. In order to set the change, a save and reboot must be done from the interface.

#### 4.1.2 Upgrading the Mesh Potato

The firmware of the Mesh Potatoes are under constant development. It is therefore advisable that the MP is running the latest version of the firmware. The process of upgrading the firmware is different from MP01 and MP02. The different methods are described below.

See the SECN User Guide for respectively MP01 and MP02 in Appendix B and C for a more detailed description of how to upgrade the firmware on the Mesh Potatoes.

#### Installing Firmware on Mesh Potato 1.0

Flashing is the process of updating or changing the firmware (SECN) on the MP01. The most common way to perform the flashing process is by using the potato-flash

application [30]. This is a specialised software application for the Mesh Potato. Potato-flash can be used regardless of previously installed firmware on the Mesh Potato [31].

1. Downloaded the 64 bit potato-flash utility from <http://download.villagetelco.org/utilities/potato-flash/potato-flash-64bit/> to the folder **/etc/local/bin**.
2. Made the potato-flash file executable by writing the following command in the Linux terminal.  

```
$ chmod +x /usr/local/bin/potato-flash-x64
```
3. Downloaded the rootfs file (**openwrt-secn1\_1-GA01-MP01-root.squashfs**) and the kernel file (**openwrt-secn1\_1-GA01-MP01-vmlinux.lzma**) from <http://download.villagetelco.org/firmware/secn/stable/mp/SECN-1.1/> to a folder we called **mp\_firmware** in our local directory.
4. Opened the terminal and wrote the following commands:

**Enter root environment:**

```
$ sudo su
```

**Turn off network manager:**

```
$ service network-manager stop
```

**Bring the interface connected to the MP up:**

```
$ ip link set eth0 up
```

**Go into the directory containing the .squashfs and .lzma files:**

```
$ cd <the directory containing the .squashfs and  
.lzma files>
```

**Assign IP-address to the interface:**

```
$ ifconfig eth0 1.1.1.1
```

Before running the potato-flash utility we made sure that the MP was unplugged from its power supply, and that the Mesh Potato was connected to our PC via an Ethernet cable.

**Executing the potato-flash utility:**

```

root@l2aether-HP-Compaq-dc7900-Small-Form-Factor:/home/l2aether/mp_firmware# potato-flash-x64 eth0 openwrt-secn1_1-GA01-MP01-root.squashfs openwrt-secn1_1-GA01-MP01-vmlinuz.lzma
Reading rootfs file openwrt-secn1_1-GA01-MP01-root.squashfs with 3276800 bytes ...
Reading kernel file openwrt-secn1_1-GA01-MP01-vmlinuz.lzma with 720896 bytes ...
Note: The device has to be connected directly (not via switch/hub)
Device detection in progress.....device detection: received ARP packet with invalid length (expected: 60): 42
device detection: received ARP packet with invalid length (expected: 60): 42
device detection: received ARP packet with invalid length (expected: 60): 42
device detection: received ARP packet with invalid length (expected: 60): 42
device detection: received ARP packet with invalid length (expected: 60): 42
device detection: received ARP packet with invalid length (expected: 60): 42
Peer MAC: 00:09:45:58:e5:0f
Peer IP : 192.168.1.20
Your MAC: 00:b3:b5:c3:ff:ee
Your IP : 192.168.1.0
Connecting to Redboot bootloader
WARNING: UNPLUGGING POWER WHILE FLASHING MIGHT DAMAGE THE BOOTLOADER
HOWEVER: IF YOU SEE NOTHING SHOWING UP BENEATH THIS LINE
FOR MORE THAN A MINUTE, START AGAIN...
A flash size of 8 MB was detected.
rootfs(0x000a0000) + kernel(0x00100000) + nvram(0x00000000) sums up to 0x007a0000 bytes
Setting IP address...
initializing partitions...
Now uploading kernel...
Sending kernel, 1408 blocks...
Flashing kernel...
Loading rootfs...
Sending rootfs, 6400 blocks...
Flashing rootfs...
Flashing process completed...
Restarting device...
root@l2aether-HP-Compaq-dc7900-Small-Form-Factor:/home/l2aether/mp_firmware#

```

**Figure 4.1: Flashing the Mesh Potato.** This figure shows the flashing process from when we first flashed our Mesh Potato 1.0

```
$ potato-flash-x64 openwrt-secn1_1-GA01-MP01-root.squashfs
openwrt-secn1_1-GA01-MP01-vmlinuz.lzma
```

Briefly after the potato-flash had been executed, dots started appearing on the screen, like shown in Figure 4.1. When these dots started appearing, we plugged the power supply back into the Mesh Potato, and the process of upgrading the firmware started.

## Installing Firmware on Mesh Potato v2.0

The process of upgrading the MP's firmware is simplified with the MP2. With the MP1, command shell must be utilized in order to do this. In contrary, with the MP2, firmware upgrade can be done from the SECN web interface (more information about the interface can be found in section 4.1.3). Extra functionality has been added to the interface from MP1 to MP2. Under "Advanced" in the interface an extra tab called "Firmware" is added, and there the firmware upgrade can be executed. When this button is pressed, a new window pops up where you have to add the firmware file. The firmware for MP02 can be found here: <http://download.villagetelco.org/firmware/secn/unstable/mp02/SECN-2.0/SECN-2.0-RC4/>.

### 4.1.3 The SECN Web Interface

Village Telco provides a SECN web interface for configuration and management of individual MPs. This web interface can be accessed by entering the IP address of the MP in a browser, this interface is shown in Figure 4.2 In order to be able to do this, the PC must be on the same subnet (exact same prefix) as the MP.

**VillageTelco** **SECN Configuration**  
 Firmware: Version 2.0 RC1-madwifi MeshPotato-1  
 Date: Sun Jan 1 00:10:56 UTC 2012

**Basic** **Advanced** **Status**

**Network**  
 IP Address: 10.130.1.20 Gateway: 10.130.1.1 Find Gateway: Test

**WiFi Access Point (WPA1)**  
 Station ID: VT-SECN-AP Passphrase: potato-potato Channel: 1

**VoIP / SIP Configuration**  
 User Name: myuser Password: \*\*\*\*  
 SIP Host: sip.myhost.com Dialout Code: #  
 SIP Enable: ☐ SIP Status: Not Registered

**Password**  
 Enter Password: Repeat Password: Set Password  
 System password has not been set

**Web Server Security and Timezone**  
 Limit IP Address: ☐ Enable SSL: ☐ Time Zone: UTC

Refresh Save Restart Asterisk Reboot

**Figure 4.2: Web interface.** Displays the web interface to the Mesh Potato.

The web interface allows the user to do alterations in the MP. The web interface allows the user to alter some key parameters. Among these are changing the IP address of the device, set up the WiFi Access Point, VoIP/SIP Configurations, Password and Web Server Security. The web interface also allows the user to do changes to more advanced settings and monitoring. To get a detailed description of the web interface see the user guide in Appendix A. In addition to this, like mentioned above, the interface has been extended and improved from MP1 to MP2. More actions can be conducted via the interface with MP2.

## 4.2 The Emergency Box

The Mesh Potatoes and Village Telcos were created to get voice and data connection to areas where services like these are non existent or too expensive for the average person. As of now the Mesh Potatoes have been set up to create small or bigger networks in villages all over the world. We want to expand this solution by looking at its mobility and its usage on the go. We want to make a box that has high adaptability, enabling it to easily be used in several different scenarios, under different conditions and by different people, all with different needs. The box have to be so easy that

there are no room for misunderstanding or mistakes. It should be a to-go box ready for any situations.

#### 4.2.1 Previous/similar work

**Go Box.** Something similar has been done before by Keith Williamson, a volunteer in the Village Telco community. His idea of utilizing the Mesh Potatoes in emergency situations started with his interest in amateur radios, and the use of radios in emergency situations. He put together a "go box" by using a waterproof Pelican 1200 case. This case contained a bracket holding the Mesh Potato, a rechargeable Li-Ion or Li-Poly battery, telephone handset and junction box to provide an on/off switch [32]. Our Emergency Box differ in some ways from the Go Box made by Williamson.

**AfrikaBurn.** AfrikaBurn is a "Burning Man" festival in Tankwa Town in the Karoo (South Africa) that is held once a year [33]. This is a festival with focus on art and freedom of expression. Instead of using money, the festival attendants are inspired to trade different types of goods with each other. A Village Telco has been established at this festival a few years now. Free-standing phone boots with Mesh Potatoes powered by solar panels has been set up at the festival area. The first year five phone boots where set up around the festival area. Since it were few numbers to call, the calling became sort of random. This gave a "ChatRoulette" like effect, only with phones instead. The second year some aspects where improved from the previous year. The boots had production MPs, no lighting and new sleeves. A netbook was brought with them, and this acted as a gateway [34, 35].

\* litt mer her? hvordan funker det? hvorfor har det blitt satt opp? er det med Internet? Om vi ikke finner dette noe sted kan vi sende en mail til Steeve.

**Military** \* høre med Arild om hvas om brukes i millitæret i dag, om det er noe liknende i det hele tatt

#### 4.2.2 Key Components

The key components of our emergency box is described in Table 4.1.

#### 4.2.3 Creating the Emergency Box

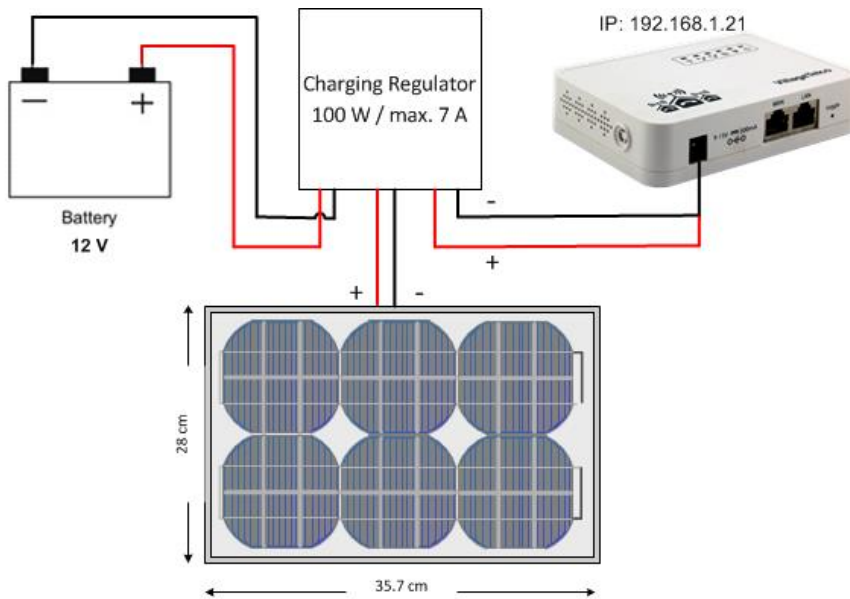
This section describes how we created the emergency box from scratch. A go-box is delivered with everything already set up. The Mesh Potato is configures and delivered with an unique IP address. In addition the suitcase contains a cd with Linux Ubuntu

**Table 4.1:** The components of Emergency Box

Component	Description and purpose
Access point for Internet	Mesh Potato v2.0
Suitcase/box	A suitcase made of high quality plastic coated with aluminium foil. Strengthened edges and corners of aluminium and steel. It has a soft-padded interior, than can be split into different rooms. Two snap locks with keys, and a solid handle for carrying. Dimensions: 455x330x152 (width, depth, height). Weight: 2,6 kg.
Power supply	A gel battery (12 V and 5 Ah). No need for maintenance. The battery acid is bound in a viscous gel. This prevents leakage, even when the battery is mounted horizontally. Long lifetime and safe to handle. The battery is fully closed, and do not need refill of battery water. No hydrogen gas or other gas might leak. When the battery is charging no gas or acid vapor is emitted, hence the battery can be placed in narrow or enclosed spaces. The battery withstands multiple discharges. The gel battery is ideal for seasonal or occasional use, since it have a slow self-discharge tempo, and a good ability to recover after deep discharging. Dimensions: 114x69x109. Weight: 2,16 kg.
Plain old telephone	A regular plain old telephone that can be connected to the MP01 via a FXS port by using a RJ-11.
Solar panel	Solar panel from Multicomp with item number: MC-SP10-GCS. Power rating: 10 W. Power Voltage Max: 17 V. Dimensions: 357x280x18
Charging regulator	Regulator for 12 V solar panel. Protects the battery from overcharging and discharge. The charging regulator is connected between the solar panel and the battery to regulate the voltage to the battery. Capacity: 100 W / max. 7 A. Overcharging protection: 14.5 V. Discharge protection: <10,5 V. Three diodes shows charging, high voltage and low battery voltage.

in case the user does not already have it installed and there would be a need for the user to configure or upgrade the MP. The case would also contain necessary cables and a detailed descriptions on how to connect to the different up-links.

**Conjoining the components of the emergency box.** The components that we used to construct the box was a solar regulator, a gel battery, a solar panel and a Mesh Potato. These components were put together as shown in Figure 4.3. The solar regulator is connected to all the components, and is used in order to not over



**Figure 4.3: The composition of the emergency box.** This figure shows how the key components in the emergency box is conjoined together.

charge the battery. The solar regulator was initially built for another type of solar panel than the one we chose to use. This means that the concomitant plugs could not be used. These plugs were cut of and we soldered on our own wires.

New wires were soldered on the solar regulator, one was connected to the power cord to the Mesh Potato, the other two fro, respectively, the battery and the solar panel. Before connecting everything together we used a multimeter in order to measure the voltage in the solar panel. We placed the solar panel under a bright lamp to see if it made any impact, which it did. We used the multimeter to check the battery as well, which was fully charged, with a voltage of a little over 13 V. We fist connected the solar panel, then the battery, and at last the Mesh Potato. In order for all to work the MP have to turn on, which it did.

**Configuring and upgrading the emergency box.** When the emergency box is delivered the Mesh Potato is already set-up and configured with an unique IP-address, and running the latest version of the SECN firmware. The descriptions will be in the box in case something would happen and there will be a need to upgrade or configure. These processes are described in sections 4.1.1 and 4.1.2.



#### 4.2.4 Battery and Charging Calculations

All the calculations done in this section is based on the components described in Table 4.1.

**Charging with solar panel.** How long it will take for the solar panel to charge the battery from fully discharged to fully charged depends on how much sun there is. The following calculations are calculated using the peak value of the solar panel (10W).

The solar panel capacity is:

$$Amp = \frac{Watt}{Volt} = \frac{10W}{17V} = 0.59Ampere$$

When charging batteries it is important to take the charging factor into account. This factor is the ratio between supplied capacity and submitted capacity. The charging factor varies depending on type of battery. For our gel battery the factor is roughly 1.2. The charging factor does not have an annotation. If the battery is completely discharged it will take:

$$Amp \times Hours = AmpHours \Rightarrow Hours = \frac{AmpHours}{Amp} = \frac{5Ah \times 1.2}{0.59A} = 10.17Hours$$

to fully charge it.

This calculation does not take into consideration that the solar panel may have to charge the battery while the MP is running. The effect from the solar panel to the battery will then decrease.

**Charge while MP01 is running.** The capacity from the solar panel will then be:

$$Amp = \frac{Watt}{Volt} = \frac{10 - 2.5W}{17V} = 0.44Ampere$$

The time it will take to fully charge the battery from fully discharged condition will then be:

$$Amp \times Hours = AmpHours \Rightarrow Hours = \frac{AmpHours}{Amp} = \frac{5Ah \times 1.2}{0.44A} = 13.6Hours$$

**Charge while MP02 is running.** The capacity from the solar panel will then be:

$$Amp = \frac{Watt}{Volt} = \frac{10 - 0.75W}{17V} = 0.54Ampere$$

The time it will take to fully charge the battery from fully discharged condition will then be:

$$Amp \times Hours = AmpHours \Rightarrow Hours = \frac{AmpHours}{Amp} = \frac{5Ah \times 1.2}{0.54A} = 11.1Hours$$

Number of sun hours per day in for example South Africa in December is approximately 14. In June number of sun hours is approximately 10.5. And off course, it can be cloudy, so these numbers are best case. This means that it is possible to fully charge the battery in the course of a day, but it may not always be the case. The following calculations shows how long the battery can provide the Mesh Potato with power from fully charged condition without the solar panel charging it at the same time.

**On fully charged battery - MP01** This calculation takes into account that the solar panel is disconnected from the battery. With the components described in Table 4.1 the number of hours the MP1 can last with fully charged battery is:

$$Amp = \frac{Watt}{Volt} = \frac{2.5W}{12V} = 0.208Ampere$$

$$Amp \times Hours = AmpHours \Rightarrow Hours = \frac{AmpHours}{Amp} = \frac{5Ah}{0.208A} = 24Hours$$

**On fully charged battery - MP02** This calculation takes into account that the solar panel is disconnected from the battery. With the components described in Table 4.1 the number of hours the MP2 can last with fully charged battery is:

$$Amp = \frac{Watt}{Volt} = \frac{0.75W}{12V} = 0.06Ampere$$

$$Amp \times Hours = AmpHours \Rightarrow Hours = \frac{AmpHours}{Amp} = \frac{5Ah}{0.06A} = 83Hours$$

#### 4.2.5 Possible Improvements

As mentioned this set-up is fairly easy and simple, but leaves room for improvements. There should be an on/off switch to not use unnecessary battery capacity. This on/off switch would be placed between the battery and the regulator. Unless a measuring instrument is available to check the voltage, there are no way for the user

to know the remaining battery time. This might be very useful in situations where communication is vital and the hours of battery lifetime are limited and should be taken into account.

The battery we used is a gel battery. They are less explosive than the lithium batteries, and are better suited to be placed into enclosed spaces that may get hot. The downside of the gel batteries is their weight, they are heavy. Since our aim is to make a case that is portable, a lighter battery would be preferable. Under the requirement of being portable a factor to keep in mind is that it is not allowed to bring lithium batteries on the plane. This means that if we chose to use a lithium battery in our solution a regular person would not be allowed to bring their emergency box on the plane. In the case of relief organizations, this might not be a problem since they transport their equipment in special planes, like a hercules.

The MP2 is a lot smaller than the first generation, and a powerful solar panel does not take that much space, so there is no need for a suitcase of the size we have purchased. A smaller case that is lighter and easier to handle would make the solution even more portable.

Our main focus was to get something that worked and that was safe, and we did not focus on aesthetics and appearance. This is therefore also an area where there are room for improvements.

### 4.3 The Process of Quick Roll-Out

As mentioned, one of our areas of focus is the process of quick roll-out. There are many aspects that can be included in order to speed up the roll-out process and make it as easy as possible. We will now present some of our ideas to meet this requirement.

#### 4.3.1 Script

A script is a list of commands that can be executed without user interaction, in other words, to automate a process. In order to connect the mesh network to Internet a list of commands have to be executed. One idea to speed up the process of setting up the network was to create a script to automate this process. One way to do this could be by creating a self-executable script that could be included in the emergency box on for example on a USB-stick. There would have to be a different script, and different USBs, for each up-link type.

\*Forklare spesifikke ting som er tatt i betrakning for å gjøre utrullingene raskest mulig. Feks scripting, manual etc.

\*Hvordan utdele telefonnummer raskt, opplæring av folk. VIKTIG. Hver boks bør ha nummer skrevet på seg allerede, og ferdig satt opp med ip-adresse.

### 4.3.2 With the MP02

The MP02 Basic does not have the option to connect to a phone. Hence the issue with telephone number distribution is irrelevant. Later in 2014 Village Telco will release a new version of the MP2, MP2-Phone. MP2-Phone will be identical to MP02 Basic, just with an FXS daughterboard. With this new version the issue of number distribution appears.

The emergency box could be delivered with several MPs, where each MP is marked with the pre-configured IP address. Since it is not possible to connect a phone to the MP there is no issue of number distribution. The box could for example contain 5 MPs, where one would be connected to an up-link, while the other ones would be strategically placed in order to spread the Internet access further.

### Distributing Numbers

When a Village Telco is set up today, telephone numbers are distributed by updating a spreadsheet with name and number to the users. These spreadsheets are printed out and delivered to everyone. This is a system that might seem cumbersome, but serves its purpose. If new nodes are added to the network or any changes are made, new sheets have to be printed out and delivered to everyone. This way of spreading telephone numbers might be more difficult with the go-box.

One option is to continue with the number distribution approach in use today. The suitcase could contain 5 MPs, all MPs are marked with its unique IP address. There will be attached a list with the IP addresses of the other Mesh Potatoes in the suitcase. When setting up the network the names can easily be filled in on each Mesh Potato. This will then be the telephone list.

An other approach would be to integrate the distribution of phone number in the web interface. A new feature could be implemented in the interface. This feature would discover the other MPs in range, also in range through other MPs. All MPs would be displayed with name of the SSID, IP address, where the last octet is the telephone number and the name of residence or user. This name could be edited by the master user or by the user themselves. Each MP in the suitcase are pre-configured and set up, they are also set up with security and a password to enter the web interface. So in order for a user to enter the web interface they have to enter the password to get access. Once inside they can see other MPs in range and also put in their name for the other MPs to see. The password and security is set up so that no other than the main user of the MP can change the name.

### 4.3.3 Get Started - How to Use the Box

\* gi en kortfattet forklaring på hva som finens i boksen og hvordan den fungerer og kan brukes, bør inneholde levetid osv.

### 4.3.4 How Connect the MP to Cabled Internet

These instructions assumes that the MP01 is pre-configured with an IP-address (192.168.1.x).

1. Make sure that the Mesh Potato is connected to a PC running Linux with an Ethernet cable.
2. The IP address of the MP is pre-configured to 192.168.1.x (where x is unique for each MP), where the x is written on the MP. In order to access the MP, the PC must be on the same subnet. To do this write in the terminal:

```
$ ifconfig eth0 up 192.128.1.120
```

3. Enter the SECN web interface by typing the IP address (192.168.1.x) of the MP01 in a browser. This is an assurance that there is contact with the MP.
4. Open Linux terminal and type in the following command:

```
$ sudo su
$ ifconfig eth0 172.31.255.253 netmask 255.255.255.252
```

5. Telnet into the MP01:

```
$ telnet 172.31.255.254
```

You have now entered the root environment of the MP01.

6. Execute udhcpd: \*Skrive hva denne kommandoen gjør

```
$ udhcpd -i eth0
```

You will get a message stating that the udhcpd process has started. This is followed by several messages stating "Sending discover...". When this appears unplug the Ethernet cable connected to the PC, and connect the MP with an Ethernet cable to cabled Internet (wall). \*Finne ut hva internett i veggen heter på engelsk.

7. Internet will now be available for the mesh network. The SSID and password for the network can be found and altered in the interface.

### 4.3.5 How Connect the MP to Internet via PC

In order to perform this set up, a PC with Linux Ubuntu and wireless Internet and a Mesh Potato 2.0 is required. The last octet of the IP address for the MP is the unique number for each MP. In the following example we use 21, but this number must be different for each MP in a network.

1. Connect the MP to the PC running Linux Ubuntu with a Ethernet cable. The Ethernet cable must be put into the LAN-port on the MP.
2. Open Linux terminal and install telnet by entering the following commands:

```
$ sudo apt-get install telnetd
$ sudo /etc/init.d/openbsd-inetd restart
```

3. The default IP address of the MPs are 10.130.1.20, so in order to access the MP, the PC must be on the same subnet. To do this write in the terminal:

```
$ ifconfig eth0 up 10.130.1.120 netmask 255.255.255.0
```

4. Open a browser on your PC and write in "10.130.1.20". The SECN Web Interface should now appear. This assures you that you have contact with the MP.

5. In the interface, change the IP address of the MP to: 192.168.1.21. Press save, and then reboot in the interface. Wait for the system to reboot.

6. Now that the IP address of the MP is changed from a local IP address, you also have to change the IP address of the ethernet port on your PC. Do this by writing this in the terminal:

```
$ ifconfig eth0 up 192.168.1.2
```

7. Now that the PC and the MP are on the same subnet, you can telnet onto the MP by writing:

```
$ telnet 192.168.1.21
```

8. Open root shell in terminal by writing:

```
$ sudo su
```

You will get prompt for password.

9. When in root shell, write:

```
$ iptables --table nat --append POSTROUTING --out-interface
eth1 -j MASQUERADE
$ iptables --append FORWARD --in-interface eth0 -j ACCEPT
$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

10. Open a editor on you PC (for example gedit). Paste the following text into a new file:

```
#!/bin/bash
ARGS=2
if [ "$(id -u)" != "0" ]; then
    echo ""
    echo "    Sorry, this script must be run as root."
    echo "    Try sudo $0 name_of_wireless_interface
    name_of_ethernet_interface"
    echo ""
    exit
fi
if [ $# -ne "$ARGS" ]; then
    echo ""
    echo "    This script sets up IP masquerading from your
    ethernet port to your wifi Internet connection."
    echo "    usage: $0 name_of_wireless_interface
    name_of_ethernet_interface"
    echo "    e.g. $0 wlan0 eth0"
    echo ""
    exit
fi
wifi=$1
eth=$2

/sbin/iptables --table nat --append POSTROUTING
--out-interface $wifi -j MASQUERADE
/sbin/iptables --append FORWARD --in-interface $eth -j ACCEPT
/bin/echo 1 > /proc/sys/net/ipv4/ip_forward
```

Save this file as "mp\_masquerade.sh" in the directory /usr/local/bin. Make the file executable by entering the following in the terminal:

```
$ sudo chmod +x /usr/local/bin/mp_masquerade.sh
```

11. Telnet into the MP and write:

```
$ telnet 192.168.1.21
$ route
$ route del default
$ route add default gateway 192.168.1.2
```

12. Now you should be able to ping IPs on the Internet from the MP. Test this by writing "ping <IP address of a Internet page>".
13. Next step is to get DNS working on the MP. In order to do this you must locate the file /etc/resolv.conf on your PC, and open in. You Copy the IP address stated in the nameserver line. Then open the SECN Web Interface, and locate the DNS field under Network under the tab Advanced. Paste the IP address into this field. Save and reboot from the interface.
14. After all this, take out the Ethernet cable from the PC and put it in the wall. WALLA, you have Wi-Fi. Go nuts!

#### **4.3.6 How Connect the MP to Cellular Network**

The following manual will describe how a user can use the emergency box to connect to a cellular network, like 3G and 4G. As well as how to use a cell phone as an access point.

#### **4.3.7 How Connect the MP to Satellite**

### **4.4 Up-Link**

Our main focus when deploying the emergency boxes, is to provide Internet to the mesh network. This because it is crucial to have the possibility to communicate with the local community and the outside world during an emergency situation. In 2011, UN declared Internet access a human right [36]. This says something about the extent of the Internet, and the importance of connectivity. In order to provide Internet to the mesh network formed by the emergency boxes, at least on the Mesh Potatoes must be connected to an access network via an uplink. An uplink connects a device or a LAN to a larger network [37]. Which type of access network that is available depends of the location. Some places there might exist a stable landline, other places not. Then an option could be to use satellite or cellular networks. It is therefore important that the emergency box has high adaptability in order to fit different scenarios. The availability of the different uplinks is not the only thing that vary. The up-link speed and the price also varies from place to place, and between the types of uplinks. In the following sections, we will look at some of the uplinks available, and how Internet access can be provided to the mesh network.



#### 4.4.1 Internet via Telephone-line

The most common way of getting Internet access is via a landline. The telephone lines are most often used for this, since they can be converted to broadband. In this way it can be used for phone calls and Internet simultaneously [38]. The line is usually in the form of twisted pairs (copper lines). These lines support broadband up to 10 Mbps, and are often in form of ADSL, or other digital subscriber line of type x (xDSL) technologies [39]. Internet via telephone lines can be provided as a stand-alone solution, or it can be provided together with television or/and phone service. The latter option is usually cheaper. Internet through landlines have a high reliability [40] in comparison to satellite and cellular network. We will now shortly describe some technologies for getting Internet access via a telephone line; dial-up Internet connection, ISDN, and DSL. Although dial-up Internet connection is practically extinct in developed countries, we will include it here due to the different application scenarios for the emergency box.

##### Dial-up Internet connection

Dial-up is an analogue technology that utilizes the telephone line. A telephone wall jack is used as a fixed point of connection, and the computer is connected to a voiceband modem. With this technology, the data is transmitted over the same frequencies used for phone calls. Hence, if you only have one telephone line, you cannot take a phone call and use Internet at the same time [41]. The absolute maximum speed is 56 kbps. Along with the digital era, better internet technologies were introduced; ISDN and DSL.

##### ISDN

Integrated Services Digital Network (ISDN) is a fixed internet connection, which also utilizes the telephone lines. When using ISDN, as with dial-up, a telephone wall jack is used as a fixed point of connection. But ISDN utilizes a ISDN terminal adapter instead of voiceband modem. This ISDN terminal adapter sends out digital signals. The data speed varies between 64 kbps - 129 kbps. The speed of the data is symmetric, which means upstream and downstream data rates are the same. In contrary to dial-up, ISDN allows voice calls and transmission of data simultaneously. ISDN is faster than dial-up, but the speed is nothing compared to the speed obtained using DSL [41].

##### DSL

Digital subscriber line (DSL) is, like the name indicates, a digital high-speed technology for Internet access that allows simultaneous voice and data transfer. Like dial-up and ISDN, DSL also run over the telephone lines. With DSL the data is not converted

between analogue and digital signals. Despite this, the signals are modulated in order to be transferred on non-voice frequencies. DSL is an always-on technology, and in this way differ from the previous technologies mentioned. Only a small part of the telephone line is used for voice signals. The DSL technology allows utilization of a unused frequency spectrum of a telephone line, hence making it possible to transmit data faster. When the voice and data signals arrive at the telephone company's local switching station, they are separated and routed differently; voice to regular telephone system and data to the ISP, and then the Internet. A connection must be within approximately 5 kilometres of a station in order for DSL to work. The speed depends on many factors. Data can be transported up to 6 Mbps (distance of approximately 2 kilometres). Relevant factors that have an impact on the speed is distance to the switching station, and the quality of the telephone line. Like mentioned earlier, there are different types of DSLs. The most common is ADSL, where the A stands for asymmetric; the downstream speed is faster than the upstream speed [41].

«««< HEAD In order to connect the Mesh Potato network to Internet via a land-line, at least one MP have to be in reach for a Internet signal. Preferably The MP should be connected with an Ethernet cable, but could also be done wireless. First the Mesh Potatoes have to be configured following the steps below. After completing the steps one can plug the Ethernet cable in in one MP and the internet is spread to the the other MPs in reach. It could be smart to change the name and the password for the different MPs. This is easily done by opening the web interface for each MP. =====

These instructions assumes that the MP01 is pre-configured with an IP-address (192.168.1.x).

1. Make sure that the Mesh Potato is connected to a PC running Linux with an Ethernet cable.
2. The IP address of the MP is pre-configured to 192.168.1.x (where x is unique for each MP), where the x is written on the MP. In order to access the MP, the PC must be on the same subnet. To do this write in the terminal:

```
$ ifconfig eth0 up 192.128.1.120
```

3. Enter the SECN web interface by typing the IP address (192.168.1.x) of the MP01 in a browser. This is an assurance that there is contact with the MP.
4. Open Linux terminal and type in the following command:

```
$ sudo su
$ ifconfig eth0 172.31.255.253 netmask 255.255.255.252
```

5. Telnet into the MP01:

```
$ telnet 172.31.255.254
```

You have now entered the root environment of the MP01.

6. Execute udhcpd: \*Skrive hva denne kommandoen gjør

```
$ udhcpd -i eth0
```

You will get a message stating that the udhcpd process has started. This is followed by several messages stating "Sending discover...". When this appears unplug the Ethernet cable connected to the PC, and connect the MP with an Ethernet cable to cabled Internet (wall). \*Finne ut hva internett i veggen heter på engelsk.

7. Internet will now be available for the mesh network. The SSID and password for the network can be found and altered in the interface.

In order to perform this set up, a PC with Linux Ubuntu and wireless Internet and a Mesh Potato 2.0 is required. The last octet of the IP address for the MP is the unique number for each MP. In the following example we use 21, but this number must be different for each MP in a network.

1. Connect the MP to the PC running Linux Ubuntu with a Ethernet cable. The Ethernet cable must be put into the LAN-port on the MP.
2. Open Linux terminal and install telnet by entering the following commands:

```
$ sudo apt-get install telnetd
$ sudo /etc/init.d/openbsd-inetd restart
```

3. The default IP address of the MPs are 10.130.1.20, so in order to access the MP, the PC must be on the same subnet. To do this write in the terminal:

```
$ ifconfig eth0 up 10.130.1.120 netmask 255.255.255.0
```

4. Open a browser on your PC and write in "10.130.1.20". The SECN Web Interface should now appear. This assures you that you have contact with the MP.
5. In the interface, change the IP address of the MP to: 192.168.1.21. Press save, and then reboot in the interface. Wait for the system to reboot.

6. Now that the IP address of the MP is changed from a local IP address, you also have to change the IP address of the ethernet port on your PC. Do this by writing this in the terminal:

```
$ ifconfig eth0 up 192.168.1.2
```

7. Now that the PC and the MP are on the same subnet, you can telnet onto the MP by writing:

```
$ telnet 192.168.1.21
```

8. Open root shell in terminal by writing:

```
$ sudo su
```

You will get prompt for password.

9. When in root shell, write:

```
$ iptables --table nat --append POSTROUTING --out-interface
eth1 -j MASQUERADE
$ iptables --append FORWARD --in-interface eth0 -j ACCEPT
$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

10. Open a editor on you PC (for example gedit). Paste the following text into a new file:

```
#!/bin/bash
ARGS=2
if [ "$(id -u)" != "0" ]; then
    echo ""
    echo "    Sorry, this script must be run as root."
    echo "    Try sudo $0 name_of_wireless_interface
    name_of_ethernet_interface"
    echo ""
    exit
fi
if [ $# -ne "$ARGS" ]; then
    echo ""
    echo "    This script sets up IP masquerading from your
    ethernet port to your wifi Internet connection."
    echo "    usage: $0 name_of_wireless_interface
    name_of_ethernet_interface"
```

```

        echo "        e.g. $0 wlan0 eth0"
        echo ""
        exit
    fi
    wifi=$1
    eth=$2

    /sbin/iptables --table nat --append POSTROUTING
        --out-interface $wifi -j MASQUERADE
    /sbin/iptables --append FORWARD --in-interface $eth -j ACCEPT
    /bin/echo 1 > /proc/sys/net/ipv4/ip_forward

```

Save this file as "mp\_masquerade.sh" in the directory /usr/local/bin. Make the file executable by entering the following in the terminal:

```
$ sudo chmod +x /usr/local/bin/mp_masquerade.sh
```

11. Telnet into the MP and write:

```

$ telnet 192.168.1.21
$ route
$ route del default
$ route add default gateway 192.168.1.2

```

12. Now you should be able to ping IPs on the Internet from the MP. Test this by writing "ping <IP address of a Internet page>".
13. Next step is to get DNS working on the MP. In order to do this you must locate the file /etc/resolv.conf on your PC, and open in. You Copy the IP address stated in the nameserver line. Then open the SECN Web Interface, and locate the DNS field under Network under the tab Advanced. Paste the IP address into this field. Save and reboot from the interface.
14. After all this, take out the Ethernet cable from the PC and put it in the wall. WALLA, you have Wi-Fi. Go nuts!

In order to connect the Mesh Potato network to Internet via a landline, at least one MP have to be in reach for a Internet signal. Preferably The MP should be connected with an Ethernet cable, but could also be done wireless. First the Mesh Potatoes have to be configured following the steps below. After completing the steps one can plug the Ethernet cable in in one MP and the internet is spread to the the

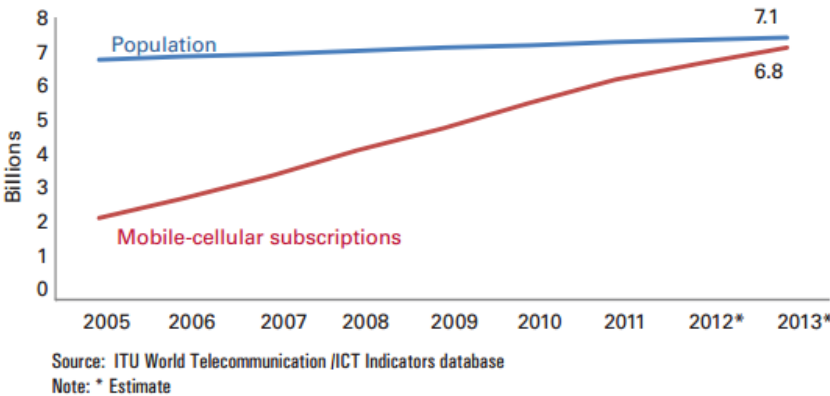
other MPs in reach. It could be smart to change the name and the password for the different MPs. This is easily done by opening the web interface for each MP. »»»>13b0e3d4e2ddfed6a8b5bf6fa70c6c0840942a6c

When connection to a wireless access point the configuration differs a little bit.

#### 4.4.2 Cellular Network Technologies

It is getting more and more common to use cellular technologies for broadband. Around 2011 the number of mobile-broadband subscriptions grew to twice as many as the number of fixed-broadband subscriptions. In developed countries it is common to have a fixed-broadband connection, and use a mobile-broadband network in addition to the fixed. In developing countries on the other hand, it is not a given that there is access to a fixed-broadband connection. Then mobile-broadband can be the only method of access available. In 2011, 90 % of the world's population had 2G coverage, and 45 % had 3G coverage [43]. By 2013, the number of mobile-cellular subscriptions had reached a high level, and were approaching the number of people in the world, like shown in Figure 4.4. From 2011 to 2013, the number of mobile-broadband subscriptions more than doubled in developing countries [42].

Through mobile network technologies, high-speed Internet access can be provided via portable devices. In order to get mobile broadband, there must be a cellular network (GSM, CDMA) service available. The key technologies when talking about mobile broadband is 3G and 4G (respectively third and fourth generation wireless networks) [44]. With 3G the average speed is 0.5 to 1.5 Mbps, and with 4G the



**Figure 4.4: Number of mobile-cellular subscriptions** The figure shows that the growth of mobile-cellular subscriptions have increased drastically during the last decade, and show that there are almost as many as there are people in the world [42].

average speed is 2 to 12 Mbps. These vary, due to different versions of each technology, underlying service etc. Like with everything else, the actual and realistic speed differ from the peak speed [45].

### Quickly Connect the Emergency Box to Cellular Network

\*Forklare hvordan man raskest mulig kobler nødboksen til akkurat denne type uplink.

#### 4.4.3 Satellite

Internet from satellites are offered by a satellite Internet provider [40]. The satellite are orbiting the Earth, and get signals from a land based Internet connection. To get Internet broadband via satellite you need a satellite dish. The main advantage of using satellite is that it provides an universally available Internet access [46]. Since it is universally available, it is fitted for use in rural regions where there exists no landlines or other options for connecting to the Internet. There also exists disadvantages with using satellite-Internet. Since it is a shared medium, privacy concerns arise, and the speed are dependent of simultaneous use. Also the connection can be affected by bad weather, unlike for a wired connection, hence it is not as reliable as cable.

### Quickly Connect the Emergency Box to Satellite

\*Forklare hvordan man raskest mulig kobler nødboksen til akkurat denne type uplink.

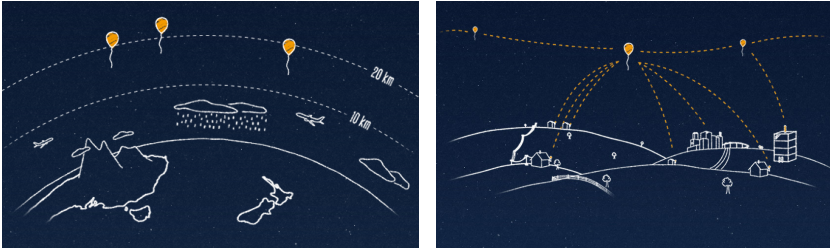
\*Normalt er det ethernet ut på disse satelittmodemene

#### 4.4.4 Summary Up-Links

**Table 4.2:** Advantages and disadvantages - Up-links [3].

Up-links	Advantages	Disadvantages
Landline/xDSL	High reliability, cost effective, good speed.	Low availability in rural areas.
Cellular networks	High availability, fitted for "on the move"-use.	Expensive, slower than xDSL.
Satellite	High availability.	Unreliable, expensive, slower than landline.

\*Forklare ulikhetene og likhetene ved å koble nødboksen til de forskjellige up-linkene.



(a) The Balloons are situated in the stratosphere. (b) Connecting to the Internet

**Figure 4.5:** Project Loon: Balloon-powered internet for everyone.

## 4.5 Future Internet Access Methods

Different methods of distributing Internet is always under development. The previous up-links described is well established, but in many parts of the world not fully developed or not affordable to the average person. The large technology companies, like Google, are experimenting with different ways of distributing Internet.

### 4.5.1 Google's Internet Balloons

The majority of the world today is not connected to the Internet. Two thirds of the worlds population does not have access. Project Loon, a Google project, is a network of high altitude balloons travelling in the stratosphere, and through this network be able to give Internet to the Entire world. Cost effective, reliable and inexpensive internet connection to everybody. The project started in June 2013 as a experiment in New Zealand [47].

The balloons are 15 meter in diameter. They travel about 20 km up in the air, in the stratosphere, twice as high as airplanes and weather, this is shown in Figure 4.5a. At this altitude there are many layers of wind, each varies in direction and speed. By regulating which wind the balloons are flying in it is possible to control their position, and steer them in the desired direction. Figure 4.5b show that people can connect to the Internet shared by the balloons by having a special antenna attached to their house. From this antenna the signal bounces to one balloon which again bounces through the other balloons and down to the Local Internet Provider on earth. This creates a network in the sky.

In order to control the altitude of the balloon, it is a specially designed control system. The system is managed remotely from the ground. By either pumping air in, or letting air out of the balloon, one can decide in what layer of air the balloon should be in. Letting air in and out is not the only way to decide whether the balloon



should go up or down, but it is the only way to do so in huge scale. A GPS is attached to each balloon in order to keep track of precise positions and see how the winds are changing. There are enormous amounts of data collected, and some of this information is given to meteorologists. The balloons are flying at the same speed as the wind.

The balloons contains specially designed antennas and radio systems. This in order to receive signals from Project Loon only, and to achieve high bandwidth over long distances. Satellites stay at the same place and at the same altitude. This means that the satellite dish can be mounted in the right direction. This is not the case with the balloons, they are in constant movement and they also vary in attitude. A fixed pointing dish would therefore not work. The antenna needs more sensitivity to an angle than it does straight up, which results in a uniform signal strength.

Since the balloons are in constant movement, it is important to make sure that there always is a balloon available and one ready to move in when one move out so that the connectivity always are available. If not this project would not be of much use. Every balloon contains information about all the other balloons in order to spread out nicely. Think of it as a flock of birds, they always look at the one next to them and space out perfectly.

The balloons are solely run on solar power. The balloon works as a communication tower in the sky. The solar panels catches the sunlight that is available during the day, as well as charging up the lithium-ion battery to last the night. In the stratosphere it is -70 degrees Celsius, these extreme cold conditions are not ideal for the lithium battery. In order to make sure that the battery does not lose its effective battery capacity it is important that the battery is kept warm. The battery pack is insulated to reflect the heat that comes from the electronics to stay warm. This is still under development.

When it comes to lifetime, the goal is that the balloons stay alive for 100 days, or 3 laps around the globe. It is important to make sure that the balloon is leak free. The material is under a lot of stress, air is constantly being pumped in and out to control the position. As well as the extreme cold and warm temperatures.

Both on the way up and down the air traffic control in the specific country have to be contacted since the balloons go through airspace. Project loon requested permission to land on Norwegian soil according to *Teknisk Ukeblad*, a Norwegian science magazine. This permission was approved [48].

The area of usage is enormous. Situations where the Internet infrastructure has not been built out, either because it is too expensive or just not possible. Emergency situations, natural disaster and other situations where the cellphone-towers are down.

Project Loon is still a project in development with extensive testing taking place. The project are taking into use what everybody have in common, namely the sky, to reach out to all the areas where access has not been possible. It is expensive to have enough balloons in the sky in order to have good enough coverage. Also the speed is not really high. And even though a specialized antenna is required to get access, this is breakthrough in order to get the whole world connected [49, 50].

Although this solution is not an option for distributing Internet as of today, it could be a good option in the near future. The solution is affordable and well compatible with our emergency box solution.

Iridium satellite consultation is a large group of satellites providing data and and voice services to specialized satellite phones. The consultation exists of 66 active satellites in orbit. Iridium are considered to be low-orbit satellites and are situated at a hight of 781 km. The Iridium network is unique in the fact that it covers the whole earth, including oceans, airways and poles. The low orbit satellite differs from the balloons in the way that the satellites travel almost 40 times higher above earth and that they stay in a fixed position over the earth [51]. To be able to utilize the satellite one would need a specialized satellite phone or a satellite dish in order to receive signal. With the balloons there are only need for an antenna to receive the signal. But then again the balloons does not offer a voice service. The balloons are intended as a cheaper, easier and simpler solution to the satellites.

## 4.6 Apple's Mesh Network

In March 2014 a new iOS app was released, FireChat. FireChat utilizes Apple's Multipeer Connectivity Framework introduced in iOS7 [52]. This app enables the possibility to chat with people "off-the-grid" [53]. Applications that communicates through this framework creates a mesh network similar to the one created by the Mesh Potatoes.

The Multipeer Connectivity framework provides support for discovering services provided by nearby iOS devices using peer-to-peer Wi-Fi, infrastructure Wi-Fi networks and bluetooth to communicate with those services. This communication could either be message-based data, streaming data and resources such as files [54]. These technologies have a short range, but this range can be greatly extended by a chain of users that creates a mesh network, see section 2.3.4 for more information about mesh networks. AirDrop is a product that have been on the market for a while and also utilizes mesh networks. The main difference is that FireChat is fully decentralized and peer-to-peer. When there are multiple users in one area FireChat relay messages in the same way as Internet, from node to node, just in this case it is from phone to phone. This, not only, enables two users to chat with each other

without Internet connection, but also far beyond Wi-Fi and Bluetooth range from each other, using the chain of users (phones). For example if Bob is connected to Alice, and Alice is connected to Carol, Bob and Carol can send messages to each other. This chain can be indefinitely long. As long as no one device goes out of Wi-Fi range, all the devices can communicate with each other.

This new framework will mainstream wireless mesh networking. This could open for a future way of spreading Internet access. This could for example be a hotel basement, cave or rural areas where there are no cellphone towers, or disaster situations where Wi-Fi or mobile broadband is no longer available. There are many benefits with the use of mesh networks. Mesh networks does not require a centralized infrastructure, there are no need for all the devices to be connected to Internet (as a router). Another benefit is that the mesh network is really easy to set up - everybody just uses the app FireChat (or similar applications like AirDrop), the network is created and everybody is connected. Simple as that!

The possibilities for this feature are enormous. Both in the creation of applications but also the area of usage. In a lot of countries Internet and mobile broadband connection is extremely expensive, people might afford a used cellphone but not the cost to connect. With this new feature, Internet connectivity can be spread through the mesh network needing only one node (phone) to have internet access. This way of spreading connectivity can open the possibility for people in rural, poorer areas like slums and small villages to stay connected. But not only the poor and rural areas can benefit from this new mesh-networking feature. Young people that does not have a phone can use an iPad or similar to talk to their friends. Or teens with restricted cell contracts can still connect with their friends, just with the help of the neighbours kids phone. Since FireChat enables communication without the use of internet it can be a useful tool to communicate privately and also to send sensitive data.

It is not only Apple that is seizing the enormous potential in main-streaming mesh networking. Google has also expressed that they are working on a home mesh network [55]. FireChat and AirDrop is just the beginning. We believe more extensive and mind blowing applications are to come.

## 4.7 Different Scenarios Where a Quick Roll-out is Necessary

Everyday there are situations all over the world that in some way affects the modern communications systems, or causes a need for one. These situations can range from big natural disasters, like the tsunami in Japan, to temporary refugee camps and IDP camps. Also more festive situations can have use of the quick roll-out communication system, for example music festivals.

#### 4.7.1 Natural Disasters

A Natural disaster is defined as; *any event or force of nature that has catastrophic consequences, such as avalanche, earthquake, flood, forest fire, hurricane, lightning, tornado, tsunami, and volcanic eruption* [56].

All over the world relief organizations are ready to help if an unexpected situation occur. These groups of people have the equipment, knowledge, experience and funding to help people in desperate need. Where are they needed? Or if they hear about the disaster and the first respond team are on place, how do they report back about the situation? How do they communicate with each other to work more effectively and help the ones in desperate need? There is no doubt that there is a need for a simple, fast, and reliable communication system.

When a natural disaster hits it is hard to know the extent of it, which again makes it difficult to predict how the communication systems would be affected. This unpredictability makes it important to always have a back-up plan to the back-up plan. History shows that cellphone service is not a reliable service during an emergency situation. During 9/11 the system became heavily overloaded, and when hurricane Katrina hit, 70% of the cell phone towers where knocked down. One might think that if they live in a big metropolitan that they would be safer, but this is not necessarily the case [57]. In addition, these examples are from the western world. The western communication systems tend to be more robust initially, compared to the ones in developing countries.

When looking at the less developed world, which is often more vulnerable to natural disasters, the situations are different. According to [58, 59] developing countries are in an larger extent affected by natural disasters than the developed world. The reason for this can be explained by the economic status of the country. Both in how the country is prepared for the disaster and in how fast they can rebuild and recuperate after a natural disaster. Developing countries often lack the infrastructure needed to quickly and efficiently provide aids to the ones affected. According to Baxter [59], a natural disaster could set back a developing country many years in development.

People are extremely dependent on communication systems when natural disasters occur. It is crucial to have the possibility to inform others of their current situation and what is needed. Time can be everything in situations like these, it can be life or death. Time can be spared if the ability to communicate without having to do it face-to-face is offered. Doing a quick roll-out of emergency boxes could be a good solution in situations like these.

Following comes two examples, respectively from a developed country, and a

underdeveloped country. The examples shows how the situation was handled, and what kind of back up communication systems that were used.

**Hurricane Sandy** Hurricane Sandy hit big parts of the Caribbean, as well as the south-east parts of the United States at the end of October 2012 [60]. As many as 25% of the citizens in the affected areas lost cell phone coverage, and even more lost electricity. Emergency communication is a challenge in natural disasters, and often leave the public without a way to call the emergency number, but also makes it difficult for first responders (as fire fighters, police, etc) to communicate. Satellite is sometimes used but is an expensive solution and they have more fixed restrictions, plus the fact that the equipment needed, phones and dishes, are expensive. No single communication system is fault free, and there always have to be a backup to the backup. Satellite was used but the phones are expensive and the lines can be oversaturated if other parties are trying to connect to the network at the same time. A small aperture terminal (VSAT) trailer was also used to act like a satellite ground station. Finding a good spot for the trailer can be tricky, it requires clear view to the sky and can not be placed too close to a large building. The Red Cross launched an emergency preparedness application for smart phones. The application had a peak in downloads right before the hurricane hit, but when the commercial wireless network failed, they had to go back to the old way of spreading information. Distributing paper files, going from house to house to check up on people's welfare, give information word-by-mouth and using bullhorn [61].

**Philippines** \*Skriver mer her

November 8 2013 the typhoon Haiyan, a powerful tropical cyclone, struck and destroyed parts of south-east Asia, in particular the Philippines. Haiyan is the strongest hurricane in wind speed ever recorded. The hurricane had the highest number for casualties, killing at least 6,268 people in the country alone [62]. International humanitarian and the government at the Philippines was warned about the storm in advance, but nobody could anticipate its viciousness. Some of the first teams on the spot was the communication experts, in order to coordinate, and make sure information was spread as desired. [63]

Clusters are groups of UN and non-UN humanitarian organisations that specialise in emergency response in areas such as water and sanitation, health, shelter, logistics, food security and agriculture. The cluster approach was applied for the first time after the 2005 earthquake in Pakistan [63].

### 4.7.2 Temporary Refugee and IDP camps

We got a better understanding of refugee and IDP camps after conducting interviews with different relief organizations (for interview with respectively CARE and Norwegian Refugee Council see section 3.3 and section 3.4). Not all refugee and IDP camps are as well established, like the ones in Dadaab. Many camps are short-term, and are therefore in more need of a temporary communication system. In this case, setting up Mesh Potatoes in the camp to provide the refugees/IDPs with Internet access is an option.

### 4.7.3 Festivals

Imagine you are at a music festival with your friends in a foreign country. There are thousands of people, and much activity. In a scenario like this there are many reasons why an Internet connection would be beneficial. You could loose your friends, have to inform your friends about something urgent, inform the staff if an emergency situation occur and so on. Since it is very expensive to send text messages/make phone calls or use mobile networks when you are abroad, it could be an idea to use Mesh Potatoes to provide the people at the festival with Internet access. This could be set up by the organizers in advance. Although this adds an extra cost to the organizers, the people attending the festival can save a lot of money by refraining from using the expensive services available on their cell phones. The organizers could add an extra fee to prize of the festival pass, and it would probably still be beneficial for the people at the festival.

### 4.7.4 Breakdown of Mobile Towers

The 10th of June 2011 one of Telenor had problems with one of its servers in Oslo. This problem caused a down time of 18 hours and affected 3 000 000 Telenor users [64]. Not only was this the biggest problem Telenor have had since they opened their mobile network in 1993, but also the longest downtime and highest number of affected users recorded in Norway. In addition to this it all happened in a period with severe flooding in big parts of the eastern Norway, and made it difficult to reach emergency numbers. The fact that the problems occurred during the flooding just made the situation much worse [65].

# Chapter 5

## Discussion





# Chapter 6

## Conclusion



# References

- [1] A. Sedghi and S. Rogers, “Unhcr 2011 refugee statistics: full data,” June 2011. <http://www.theguardian.com/news/datablog/2011/jun/20/refugee-statistics-unhcr-data>, accessed 24.03.2014.
- [2] A. Sedghi, “Unhcr 2012 refugee statistics: full data,” June 2013. <http://www.theguardian.com/news/datablog/2013/jun/19/refugees-unhcr-statistics-data>, accessed 20.03.2014.
- [3] S. Cope, “Internet connection and access methods,” 2011. <http://www.steves-internet-guide.com/connect-methods/>, accessed 31.03.2014.
- [4] J. Dempsey, “The mesh potato network,” 2008. <http://ictupdate.cta.int/en/Feature-Articles/The-mesh-potato-network>, accessed 26.02.2014.
- [5] Village Telco, “Village telco workshop,” 2008. <http://villagetelco.org/2008/07/village-telco-workshop/>, last accessed 24.02.2014.
- [6] Village Telco, “The origin of the mesh potato,” Last edited: 2013. <http://villagetelco.org/2008/06/the-origin-of-the-mesh-potato/>, accessed 21.02.2014.
- [7] Village Telco, “Background village telco,” Last edited: 2013. <http://wiki.villagetelco.org/Background>, accessed 20.02.2014.
- [8] Village Telco, “Mesh potato,” 2013. <http://store.villagetelco.com/mesh-potatoes/mesh-potato.html>, last accessed 20.02.2014.
- [9] Wikipedia, “Binary blob,” Last modified December 2013. [http://en.wikipedia.org/wiki/Binary\\_blob](http://en.wikipedia.org/wiki/Binary_blob), accessed 05.03.2014.
- [10] D. Rowe, “The mesh potato part 1,” 2008. <http://www.rowetel.com/blog/?p=70>, accessed 26.02.2014.
- [11] OpenWrt Wiki, “About openwrt.” <http://wiki.openwrt.org/about/start>, accessed 03.03.2014.
- [12] Indiana University, “What is telnet?.” <http://kb.iu.edu/data/aayd.html>, accessed 25.04.2014.

- [13] Indiana University, “What is telnet/ssh?.” <https://service.futurequest.net/index.php?/Knowledgebase/Article/View/31>, accessed 25.04.2014.
- [14] J. Hoebeke, I. Moerman, B. Dhoedt, and P. Demeester, “An overview of mobile ad hoc networks: Applications and challenges,” 2004.
- [15] X. Hong, K. Xu, and M. Gerla, “Scalable routing protocols for mobile ad hoc networks,” *IEEE*, 2002.
- [16] J. Wang, B. Xie, and D. P. Agrawal, *Journey from Mobile Ad Hoc Networks to Wireless Mesh Networks*, pp. 1–30. Springer London, 2009.
- [17] B. D. Shivahare, C. Wahi, and S. Shivhare, “Comparison of proactive and reactive routing protocols in mobile adhoc network using routing protocol property,” *International Journal of Emerging Technology and Advanced Engineering*, 2012.
- [18] A. Neumann, C. Aichele, M. Lindner, and S. Wunderlich, “Better approach to mobile ad-hoc networking (b.a.t.m.a.n.),” 2008. <http://tools.ietf.org/html/draft-wunderlich-openmesh-manet-routing-00>, accessed 21.02.2014.
- [19] Freie Universität Berlin, “Better approach to mobile ad hoc networking (b.a.t.m.a.n.).” <http://www.des-testbed.net/content/better-approach-mobile-ad-hoc-networking-batman>, accessed 24.02.2014.
- [20] Village Telco, “Spud – simple unified dashboard for mesh networks.” <http://villagetelco.org/2011/06/spud-simple-unified-dashboard-for-mesh-networks/>, accessed 26.02.2014.
- [21] The UN Refugee Agency, “What is a refugee?.” [http://www.unrefugees.org/site/c.lfIQKSOWFqG/b.4950731/k.A894/What\\_is\\_a\\_refugee.htm](http://www.unrefugees.org/site/c.lfIQKSOWFqG/b.4950731/k.A894/What_is_a_refugee.htm), accessed 20.03.2014.
- [22] Refugees International, “Somalia.” <http://refugeesinternational.org/where-we-work/africa/somalia>, accessed 24.03.2014.
- [23] Refugees International, “Iraq.” <http://www.refugeesinternational.org/where-we-work/middle-east/iraq>, accessed 24.03.2014.
- [24] Inveneo, “How better connectivity can help dadaab, the world’s largest refugee camp,” 2012. <http://www.inveneo.org/2012/06/how-better-connectivity-can-help-dadaab-the-worlds-largest-refugee-camp/>, accessed 10.03.2014.
- [25] Care, “Dadaab refugee camps, kenya.” <http://care.org/emergencies/dadaab-refugee-camp-kenya>, accessed 11.03.2014.
- [26] Wikipedia, “Nethope,” Last edited: 13 May 2013. <http://en.wikipedia.org/wiki/NetHope>, accessed 11.03.2014.
- [27] Inveneo, “Dadaabconnect.” <http://www.inveneo.org/projects/dadaabconnect/>, accessed 11.03.2014.

- [28] M. Maasilta, “Outsiders or active citizens? the role of oral and mediated communication in african refugee camps,” ?
- [29] Village Telco, “Choosing a firmware for the mp.” <http://villagetelco.org/get-started/choosing-a-firmware-for-the-mp/>, accessed 09.04.2014.
- [30] Village Telco, “Flash your mesh potato.” <http://villagetelco.org/get-started/flash-your-mesh-potato/>, accessed 09.04.2014.
- [31] Village Telco, “Installing vt secn firmware.” [http://wiki.villagetelco.org/Installing\\_VT\\_SECN\\_Firmware#Using\\_Potato\\_Flash\\_software](http://wiki.villagetelco.org/Installing_VT_SECN_Firmware#Using_Potato_Flash_software), accessed 09.04.2014.
- [32] K. Williamson, “Guest post: Adapting mesh potatoes for emergency work.” <http://villagetelco.org/2011/11/guest-post-adapting-mps-for-emergency-work/>, accessed 02.04.2014.
- [33] AfrikaBurn, “What is afrikaburn?” <http://www.afrikaburn.com/about/what-is-afrikaburn>, accessed 05.05.2014.
- [34] S. Song, “Africa burns for a village telco.” <http://manypossibilities.net/2010/05/africa-burns-for-a-village-telco/>, accessed 05.05.2014.
- [35] D. Carman, “Afrikaburns again for a village telco.” <http://villagetelco.org/2011/05/afrikaburns-again-for-a-village-telco/>, accessed 05.05.2014.
- [36] D. Kravets, “U.n. report declares internet access a human right.” <http://www.wired.com/threatlevel/2011/06/internet-a-human-right/>, accessed 14.03.2014.
- [37] B. Mitchell, “What is an uplink(port)?.” <http://compnetworking.about.com/od/homenetworking/f/uplink-port.htm>, accessed 14.03.2014.
- [38] Digital Unite, “How to connect to internet.” <http://digitalunite.com/guides/using-internet-0/connecting-internet/how-connect-internet>, accessed 27.03.2014.
- [39] J. A. Audestad, *Technologies and Systems for Access and Transport Networks*. Artech House, Inc., 2008.
- [40] A. Chianis, “Cable vs. satellite — which internet connection serves your business best?.” <http://www.businessbee.com/resources/news/technology-buzz/cable-vs-satellite-internet-connection-serves-business-best/>, accessed 14.03.2014.
- [41] J. J. Parsons and D. Oja, *New Perspectives on Computer Concepts 2010: Comprehensive*, pp. 313–315. Course Technology, 2010.
- [42] International Telecommunication Union, “The world in 2013: Ict facts and figures.” <http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013-e.pdf>, accessed 28.03.2014.
- [43] International Telecommunication Union, “The world in 2011: Ict facts and figures.” <http://www.itu.int/ITU-D/ict/facts/2011/material/ICTFactsFigures2011.pdf>, accessed 28.03.2014.

- [44] B. Mithcell, “What is mobile broadband?” <http://compnetworking.about.com/od/internetaccessbestuses/f/what-is-mobile-broadband.htm>, accessed 28.03.2014.
- [45] Diffen, “3g vs 4g.” [http://www.diffen.com/difference/3G\\_vs\\_4G](http://www.diffen.com/difference/3G_vs_4G), accessed 28.03.2014.
- [46] E. S. Cohen, ed., *Broadband Internet: Access, Regulation and Policy*, pp. 80–81. 2007.
- [47] Project Loon, “Projet loon: Balloon-powered internet for everyone.” <http://www.google.com/loon/>, accessed 27.03.2014.
- [48] R. Ramsdal, “Google fikk ja til å sende internett-ballonger over norge,” March 2014. <http://www.tu.no/it/2014/03/22/google-fikk-ja-til-a-sende-internett-ballonger-over-norge>, accessed 23.03.2014.
- [49] Project Loon, “Project loon,” 2013-2014. <http://www.youtube.com/user/ProjectLoon>, accessed 28.03.2014.
- [50] A. B. SMEDSRUD, “Disse ballongene skal gi internett til hele verden,” June 2013. <http://www.hardware.no/artikler/disse-ballongene-skal-gi-internett-til-hele-verden/134539>, accessed 28.03.2014.
- [51] Wikipedia, “Iridium satellite constellation.” [http://en.wikipedia.org/wiki/Iridium\\_satellite\\_constellation](http://en.wikipedia.org/wiki/Iridium_satellite_constellation), accessed 09.04.2014.
- [52] M. Elgan, “How an under-appreciated ios 7 feature will change the world,” March 2014. <http://www.cultofmac.com/271225/appreciated-ios-7-feature-will-change-world/>, accessed 25.03.2014.
- [53] FireChat, “app store: Firechat.” <https://itunes.apple.com/us/app/firechat/id719829352?mt=8&ign-mpt=uo%3D8>, accessed 27.03.2014.
- [54] Apple, “About multipeer connectivity,” September 2013. <https://developer.apple.com/library/ios/documentation/MultiPeerConnectivity/Reference/MultiPeerConnectivityFramework/Introduction/Introduction.html>, accessed 25.03.2014.
- [55] M. Elgan, “Why google is working on home mesh networking,” March 2014. <http://www.eweek.com/cloud/why-google-is-working-on-home-mesh-networking.html>, accessed 27.03.2014.
- [56] Dictionary.com, “Natural disaster.” <http://dictionary.reference.com/browse/natural+disaster>, accessed 19.03.2014.
- [57] S. Kelly, “What is the best shtf/disaster communication?.” <http://graywolfsurvival.com/2716/ham-radio-best-shtfdisaster-communication/>, accessed 01.04.2014.
- [58] D. Smithfield, “Haiti’s uphill battle: Developing countries struggle with natural disasters,” October 2013. <http://www.refugeesinternational.org/blog/haiti%E2%80%99s-uphill-battle-developing-countries-struggle-natural-disasters>, accessed 09.04.2014.

- [59] Aurecon, “360° natural disaster,” 2011. [http://issuu.com/aurecon/docs/aurecon\\_360\\_issue3?mode=embed&layout=http%3A//skin.issuu.com/v/light/layout.xml&showFlipBtn=true](http://issuu.com/aurecon/docs/aurecon_360_issue3?mode=embed&layout=http%3A//skin.issuu.com/v/light/layout.xml&showFlipBtn=true), accessed 09.04.2014.
- [60] Wikipedia, “Orkanen sandy.” [http://no.wikipedia.org/wiki/Orkanen\\_Sandy](http://no.wikipedia.org/wiki/Orkanen_Sandy), accessed 31.03.2014.
- [61] V. Insinna, “Natural disasters uncover ongoing emergency communications problems,” June 2013. <http://www.nationaldefensemagazine.org/archive/2013/January/Pages/NaturalDisastersUncoverOngoingEmergencyCommunicationsProblems.aspx>, accessed 28.03.2014.
- [62] Wikipedia, “Typhoon haiyan.” [http://en.wikipedia.org/wiki/Typhoon\\_Haiyan](http://en.wikipedia.org/wiki/Typhoon_Haiyan), accessed 01.04.2014.
- [63] M. Tran, “Typhoon haiyan disaster response: how the relief effort worked,” February 2014. <http://www.theguardian.com/global-development/poverty-matters/2014/feb/07/typhoon-haiyan-disaster-response-philippines-relief-effort>, accessed 01.04.2014.
- [64] Wikipedia, “Liste over nedetid i mobilnett i norge.” [http://no.wikipedia.org/wiki/Liste\\_over\\_nedetid\\_i\\_mobilnett\\_i\\_Norge](http://no.wikipedia.org/wiki/Liste_over_nedetid_i_mobilnett_i_Norge), accessed 19.03.2014.
- [65] Nettavisen.no, “Telenor: Feilen i mobilnett er rettet.” <http://www.nettavisen.no/nyheter/3168897.html>, accessed 19.03.2014.





# Appendix

## Interview with Care



This appendix contains the summary from the interview conducted on Mary Muia (CARE International in Kenya | Program Assistant Refugee Assistance Programme | Dadaab).

1. Approximately how many people are there in the Dadaab refugee camp? And how long have it been in operation?

The Dadaab complex of refugee camps, considered the world's largest, was created in 1991 by the Government of Kenya and UNHCR to host Somali refugees displaced by civil war. Over the years, the camps have also hosted other nationalities from the Horn of Africa, the Great Lakes and East Africa regions but they constitute less than two percent of the camp population. The original camps were Dagahaley, Ifo and Hagadera and were intended to host 90,000 refugees. However, in 2011, there was an influx of new refugees from Somalia due to severe drought and new camps were created; Ifo 2 and Kambioos, to cater to the over 175,000 new arrivals and at the peak of the influx in 2011, the camps hosted more than 463,000 refugees, including some 10,000 third-generation refugees born in Dadaab to refugee parents who were also born there. However, in 2013, UNHCR and its partners conducted a verification exercise to ascertain the current population since some of those who had arrived in 2011 due to the famine had returned home. As at February, 2014, the current population stands at 369,294.

2. How do you connect and communicate with the outside world?

CARE as an organization has invested in communication systems in liaison with Internet Service Providers in the capital city of Nairobi who ensure that all staff have access to internet for both official and social usage.

3. How are the communication inside the camp (communication flow)?

Several telecommunication firms in Kenya have put up their machinery in the area thus there is access to both mobile communication and access to internet services. There are also radio station services and access to digital televisions. CARE uses telephone services to reach out to refugee staff (50%) of the refugees have access to mobile phone services - either owned or through a bureau) posters and radio to reach out to its beneficiary population. In addition, there is word of mouth done through loud speakers during major gatherings like food distribution days and also road shows within the camps.

4. How does the refugees receive information?

As 3 above.

5. Can you explain what happens when a new person enters the camp?

Upon arrival, a new refugee would report to a UNHCR reception desk whereby they are given temporary registration pending full registration and location of their relatives is they have any already in the camp. UNHCR fully briefs the new arrival on all the services available and which Agency is handling what service. Immunizations, medical attention, emergency food supply, tarpaulins, sleeping mats, jerrycans for fetching water and kitchen sets are issued to such new arrivals to help them start their new lives in the camps. UNHCR then hands over the new arrivals to the respective Agency doing camp management in the specific camp they are allocated so that they can be shown where to pitch their tents. The camps are well demarcated into numbered sections and blocks thus at any given time, UNHCR would inform you where a particular refugee resides and the family size. Each Agency working in Dadaab has their own mode of communicating the services they provide to their target beneficiaries. However, UNHCR holds regular meetings with the refugee leaders of each respective camp whereby information is shared with them for dissemination to the entire refugee population.

6. What are the biggest challenges in a refugee camp?

Lack of enough space to accommodate everyone and lack of enough funds to take care of all the needs of the refugees.

7. What is the biggest challenge when it comes to communication/information spreading in the refugee camp?

Language barrier between the humanitarian staff and the refugees since many of the staff do not speak/understand the Somali language while 95.6% of the refugees are Somali. Internet and telephone service outages are also common in the area and response by the service providers sometime take a while.

8. What means of communication do you use in the refugee camp?

Mobile phones and computers for both telephone and internet access. Radios and television services.

9. We have the impression that there are not many telecom providers offering telecommunication services in Africa, and hence little competition. Which in general makes the prices higher. How does the ones living in the camp afford to have a phone?

(a) There are two main telecom provides here so yes, little competing thus high rates (b) Many refugees who have been here for over a long period of time have established small scale business (some supported by the NGO's i.e. IGA's (Income Generating Activities) thus make some little profit. Others have established business through the support of their relatives who have been resettled in other countries thus send them some cash while others who may have been businessmen back in Somalia made it to take some of the cash they had at the time of fleeing their country.

10. Is it "Internet cafes" that people have to pay to be able to use?

Yes some refugees have set up small internet cafes in the markets thus people who need the services have to pay for it. CARE like other NGO's here has Community Development Projects which include ICT training where we train the youth on ICT and upon successful completion, we support them by providing them with start-up kits to establish their own small cafes for both business and training others youth.

11. How long does it take to set up a communication system?

N/A since I am not a technical person

12. Do you use video surveillance?

No

13. Have you heard of something called Freedom fone?

No

14. Have you heard of the company Village Telco?

No

15. Do you have anything else to add that can be of interest for our master thesis?

No



# Appendix

## SECN-1.1 User Guide



**Village Telco  
Small Enterprise / Campus Network  
SECN-1.1**

**User Guide**





SECN User Guide by T L Gillett is licensed under a  
[Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/).  
Based on a work at [www.villagetelco.org](http://www.villagetelco.org).

### Acknowledgements

This work would not have been possible without the contributions of many people associated with Village Telco.

In particular I would like to acknowledge the considerable contributions made by Elektra both in providing technical guidance and in building the software, as well as writing the text for sections of this manual.

Much input has also been provided by Keith Williamson, particularly in relation to development of the Softphone Support and DHCP features, and in building many test versions of the firmware.

I would also like to acknowledge the ongoing support and encouragement provided by Steve Song as a founder of the Village Telco project.

**Note:** This draft document is intended to be read in conjunction with SECN-1.1 firmware.

## **Table of Contents**

1. Introduction.....	4
2. A Simple Mesh Set Up.....	5
3. Example Networks.....	6
4. Setting Up MP Devices.....	8
4.1 Installing the SECN Firmware.....	8
4.2 Minimum Set-up.....	12
4.4 Set-up Using SECN Web Interface.....	13
4.5 Advanced Set-up.....	21
5. Overview of SECN Operation .....	24
5.1 IP Address Range for MPs.....	24
5.2 Batman-Advanced Operation.....	25
5.3 Telephony Operation.....	28
5.4 Asterisk Operation.....	29
5.5 Softphone Support.....	33
5.6 USB Extended File System .....	36



## **1. Introduction**

The Small Campus Enterprise Network (SECN) firmware is designed to allow a collection of Mesh Potato (MP) and similar devices (eg various TP-Link devices) with firmware based on OpenWrt, to provide a data and telephony network for a small campus or enterprise.

The intended use is typically for a small/medium size organisation which needs to set up a number of workpoints spread over a limited geographic area, with workpoints being equipped with a telephone and a networked PC, and to do this wirelessly without using conventional LAN cabling.

On a slightly larger scale, the system may also be used to provide networking for a small community, with shared access to network resources such as web server, file server and Internet access.

The meshed devices utilise an OSI Layer 2 protocol (batman-adv) and collectively simply act as one large switch, transparently connecting all the attached devices together.

Each MP device provides a telephone connection, an Ethernet cable connection, and a WiFi Access Point. TP devices provide mesh nodes without the telephone connection. PCs and other network devices may be connected to the Ethernet port of a mesh device, or connect wirelessly to the WiFi Access Point of each node.

The WiFi Access Point in each mesh node is encrypted with WPA by default in order to provide some protection from abuse of the data network as long as the pass phrase/key is kept confidential.

The Access Points may be configured to use the same SSID and password, in which case the WiFi 'cell' will effectively cover the same area as the mesh, and WiFi client devices will 'roam' throughout the cell. Alternatively, the Access Points may be individually configured so as to provide discrete WiFi cells.

If one or more of the mesh nodes is connected via its Ethernet port to a LAN with a router / DHCP server and Internet access, any device connected either by Ethernet cable to an MP or by WiFi, will be able to acquire a DHCP address on the LAN and connect to the Internet via the router.

Similarly, networked devices such as printers or storage devices may be attached to the LAN via a mesh node device. All attached devices will appear on the LAN and will be visible to each other.

Each MP device provides a telephone port which may be called from another MP telephone by dialling the IP address of the required device. Abbreviated dialling is also supported so that a call may be made by dialling just the last octet of the required IP address.

Support is also provided for Softphone applications running on smartphones, PCs or other devices.

To use telephony off the local mesh, individual mesh node devices can be configured to access a SIP/VoIP Service Provider account for outgoing and incoming calls.

Configuration and management of individual mesh node devices is possible via telephone IVR commands (MP only), browser or terminal sessions with access to the underlying OpenWrt Linux operating system and software.

## 2. A Simple Mesh Set Up

In this simple mesh network we will set up a network of two MP devices so that phone calls can be made between them, then connect one MP to a Local Area Network with Internet access so that a laptop can connect wirelessly to the virtual Access Point and access the LAN and Internet.

**Step 1.** Flash the MP devices to the SECN firmware.  
(See following section for details of how to flash the devices.)

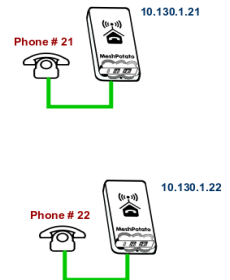
**Step 2.** Set the unique IP address for each MP device.

When the MP devices are rebooted, connect a telephone.  
Lift the receiver and check for dial tone.

Dial **2662**, enter the pin **1234**, and when the announcement has finished dial **21**. Wait to hear the number being read back, then reboot the device when prompted.

Repeat the process with the second MP, but dial **22** and wait for it to reboot.

The MP devices are now set to IP addresses **10.130.1.21** and **10.130.1.22** respectively. It may be useful to label the devices as '21' and '22'



**Step 3.** Make a phone call.

After the MP devices have fully rebooted (allow a couple of minutes after the WiFi light starts to flash), pick up the phone on the '21' MP, check for dial tone and dial **22**. The other phone should start to ring after a few seconds. Repeat the other way around.

**Step 4.** Attach the mesh network to your LAN.

Connect the MP '21' to a spare port on your router with an Ethernet cable. The diagram shows the LAN using an IP address range of 192.168.1.xxx, but the actual range used will not matter

**Note:**— *Because the mesh utilises an OSI Layer 2 protocol, it will work with any LAN address range. The MP addresses do **\*not\*** have to be in the same subnet as the LAN in order for the mesh to carry LAN data traffic.*

**Step 5.** Attach a laptop via WiFi.

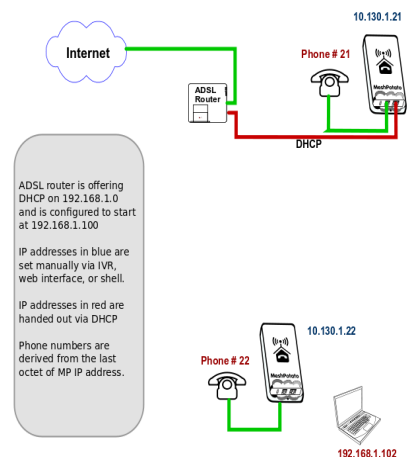
Your laptop should be able to see a WiFi Access Point called **VT-SECN-AP** secured with WPA encryption. Connect to this Access Point with a WPA password of '**potato-potato**' and using Automatic assignment of IP address (DHCP).

Your laptop should acquire an IP address in the range offered by your LAN router, and you should be able to access the Internet.

You should be able to make calls between the MP devices while accessing the Internet on the laptop.

You can connect another PC to the '22' MP using an Ethernet cable and it will similarly acquire an IP address from the router.

The laptop and PC should be able to access any other devices on the LAN, such as printers or network storage devices just as if they were connected directly to the LAN.



### 3. Example Networks

Following are examples of practical networks built around MP devices operating in a mesh.

#### Network 1

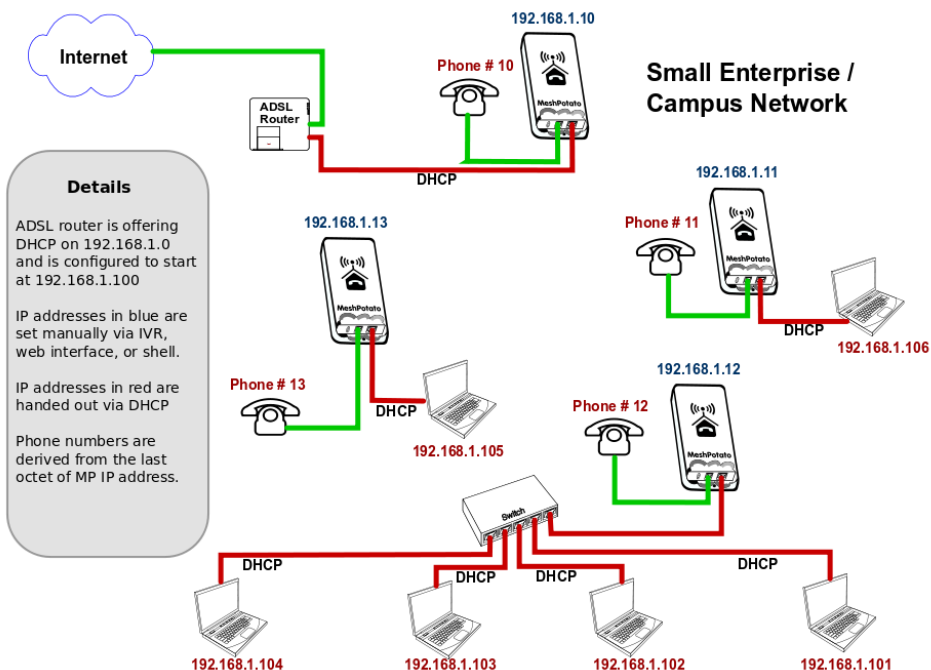
In this network, MP devices have been assigned static IP addresses that are part of the LAN address space, 192.168.1.xxx, and appropriate Gateway and DNS addresses.

This means that the MP administration interfaces (SECN web interface and *ssh* command line) will be accessible from any workstation connected to the LAN.

When a workstation is attached to the mesh network either by Ethernet cable or WiFi, it will acquire an IP address from the router in exactly the same manner as if it was connected directly to the router.

The router address space must be managed so that there is no conflict between the statically assigned MP addresses and those for any other device on the network. In this example the router offers DHCP addresses starting at 192.168.1.100, while the MPs have been assigned static addresses below this range.

Phone calls may be made between MP devices either by dialling the last octet of the required MP address (e.g. '10'), or by dialling the full address (e.g. '192\*168\*1\*10').



## Network 2

In this network, MP devices have been assigned static IP addresses that are not part of the LAN address space. Instead they have been assigned IP addresses in the default address space 10.130.1.xxx.

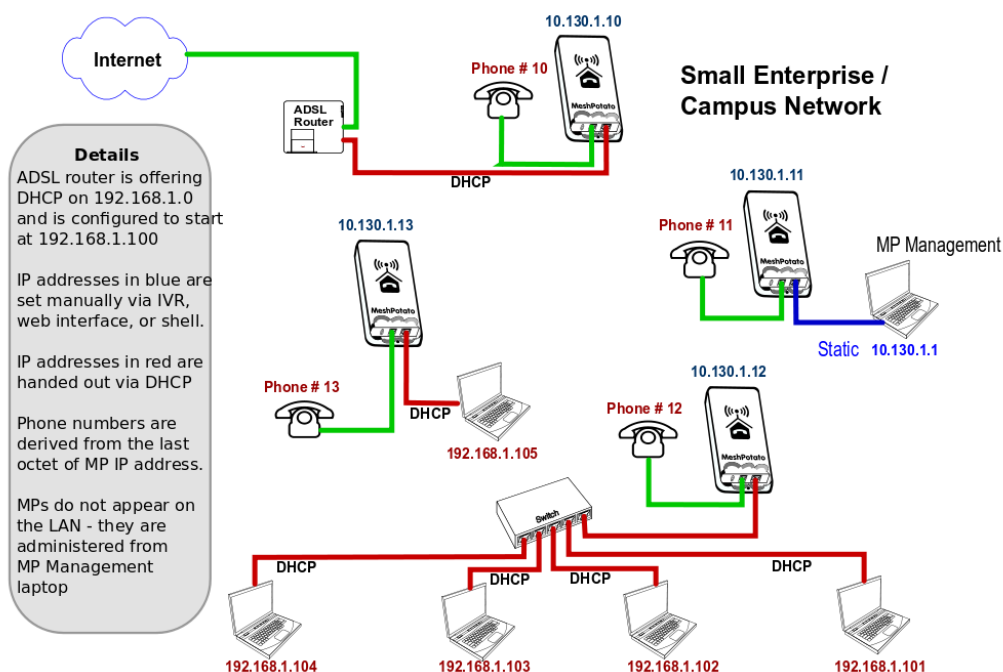
This means that the MP administration interfaces (SECN web interface and **ssh** command line) will not be accessible from workstations connected to the LAN with IP addresses assigned in the LAN address space.

Administration of the MP devices may be undertaken from a workstation assigned a static address in the same range as the MP devices and attached via Ethernet cable or WiFi to any MP device in the network.

When a workstation is attached to the network either by Ethernet cable or WiFi, it will acquire an IP address from the router in exactly the same manner as if it was connected directly to the router.

In this example there is no need to manage the LAN address space to allow for the MP addresses as they are allocated in a completely different address space (10.130.1.xxx).

Phone calls may be made between MP devices either by dialling the last octet of the required MP address (e.g. '10'), or by dialling the full address (e.g. '10\*130\*1\*10').



## 4. Setting Up MP Devices

This section describes how to set up MP or TP devices for use on your mesh network.

The first step is to install the SECN firmware on the MP or TP device.

After installing the firmware, three different methods are available to configure the device:

- **Minimum Setup** using telephone Interactive Voice Response (IVR – MP only)
- **Basic Setup** using the SECN web browser interface.
- **Advanced Command Line Setup** using a *ssh* terminal session and command line.

### 4.1 Installing the SECN Firmware

If you have purchased a new MP device, it may be delivered from the factory with VT firmware version rv233 installed. To operate with the SECN configuration, you will need to flash the MP with the appropriate SECN firmware. Similarly you may wish to upgrade the SECN firmware version.

There are two methods for installing the firmware:

- Use the **Potato-Flash** utility on a Linux PC connected to the MP via Ethernet cable.
- Use the **sysupgrade** utility from the command line on the MP/TP device.

Using the **sysupgrade** utility has the advantage that the firmware can be installed on the MP/TP 'over the air' without the need to connect with an Ethernet cable, which may avoid the need to physically recover the device from an installation.

**NOTE:** *Early MP devices may not be immediately suitable for flashing with the sysupgrade utility until they have been flashed at least once with the potato-flash utility. This is due to an incorrect memory layout from a different flashing program. If possible, use the potato-flash utility as the preferred way to flash the MP device.*

To check the status of your MP, run the command:

```
cat /proc/mtd
```

The correct output looks like:

```
dev:   size   erasesize  name
mtd0: 00030000 00010000 "RedBoot"
mtd1: 000b0000 00010000 "vmlinux.bin.l7" Note that mtd1 contains the Linux kernel
mtd2: 006f0000 00010000 "rootfs"
...
```

An incorrect output may look like this:

```
dev:   size   erasesize  name
mtd0: 00030000 00010000 "RedBoot"
mtd1: 006f0000 00010000 "rootfs" Note that mtd1 does not contain the Linux kernel
mtd2: 00410000 00010000 "rootfs_data"
...
```

## Installing MP Firmware with the Potato-Flash Utility

These instructions assume that you are running Ubuntu or other Linux distribution on your PC. If this is not the case, use one of the other methods to flash the device.

### 1. Set up the *potato-flash* application on your PC

Download the potato-flash file from:

<http://villagetelco.org/download/utilities/>

Save the file into */usr/local/bin*

Make the file executable:

***chmod +x /usr/local/bin/potato-flash***

### 2. Download the firmware

Download the required firmware from:

<http://villagetelco.org/download/firmware/secn/>

Download the *.squashfs* and *.lzma* files for the required firmware version and save to a working directory.

### 3. Set up networking on your PC

This step will ensure that *potato-flash* has proper access to the PC Ethernet network port.

Connect the MP directly to your PC with an Ethernet cable **with the MP power off**.

In Ubuntu Gnome desktop, right click on the Network Manager icon and deselect ***Enable Wireless***

Left click on the Network Manager icon and ***Disconnect*** any ***Wired Network*** that is active.

### 4. Flash the MP

Following is a brief description of the flashing process. Refer to the general instructions in ***Upgrading Mesh Potato Firmware HowTo*** on the Village Telco Wiki for more detail.

- Connect the MP directly to your PC with an Ethernet cable with the MP power **off**
- Execute potato-flash:  
***\$ sudo potato-flash eth0 <filename>.squashfs <filename>.lzma***  
 Assuming the Ethernet port on the PC is *eth0*.  
 Note that the order of the *.squashfs* and *.lzma* files is mandatory in the command.
- Enter your password when prompted.
- Wait for the program to start looking for the MP device - a series of dots will appear on the screen.
- Switch the power **on** to the MP.
- Wait for the flashing process to complete and for the MP to fully restart.  
***This may take a couple of minutes.*** This is a good time to have a coffee.
- Wait for three minutes after the MP WiFi led starts to flash to ensure that flashing process is complete. Some early MP devices may take quite a long time (10mins +) to load and flash.

### **Sample MP Potato Flash Session**

***\$ sudo potato-flash eth0 openwrt-atheros-root-rv238.squashfs openwrt-atheros-vmlinux-rv238.lzma***

Reading rootfs file openwrt-atheros-root-rv238.squashfs with 3801088 bytes ...

Reading kernel file openwrt-atheros-vmlinux-rv238.lzma with 720896 bytes ...

Note: The device has to be connected directly (not via switch/hub)

Device detection in progress.....

***<<< Turn the power to the MP device ON at this point >>>***

.....device detection: non-arp packet received..

Peer MAC: 00:09:45:58:1c:e7

Peer IP : 192.168.1.184

Your MAC: 00:ba:be:ca:ff:ee

Your IP : 192.168.1.0

Connecting to Redboot bootloader

WARNING: UNPLUGGING POWER WHILE FLASHING MIGHT DAMAGE THE BOOTLOADER

HOWEVER: IF YOU SEE NOTHING SHOWING UP BENEATH THIS LINE

FOR MORE THAN A MINUTE, START AGAIN...

A flash size of 8 MB was detected.

rootfs(0x006a0000) + kernel(0x00100000) + nvram(0x00000000) sums up to 0x007a0000 bytes

Setting IP address...

Initializing partitions...

Now uploading kernel...

Sending kernel, 1408 blocks...

Flashing kernel...

Loading rootfs...

Sending rootfs, 7424 blocks...

Flashing rootfs...

Flashing process completed...

Restarting device...

## Installing with the *sysupgrade* Utility

To install with the *sysupgrade* utility on the MP or TP device, it is necessary to copy the required *.img* file to the MP/TP using the *scp* command from within a *ssh* session on your PC. You may also use *sftp* to browse the unit's file system in Nautilus or with WinSCP.

An MP/TP device flashed with SECN firmware will only provide terminal access via *ssh* by default using the login account of *root* once the system password has been set..

If you are flashing a new TP device running the original factory firmware, you will need to use the '*factory*' version of the firmware *.img* file, rather than the one *sysupgrade* version of the *.img* file. This is required only for the first time the device is flashed to the VT SECN firmware. Use the IP address and web interface of the manufacturer's firmware to load the new firmware file.

If you are re-flashing a TP device that already has the VT SECN or other OpenWrt firmware version loaded, follow the process outlined below using the *sysupgrade* version of the firmware.

If you are using a new MP it will operate with IP addresses of 10.130.1.20 (LAN) and 172.31.255.254 (Fallback). To use one of these addresses, configure your PC Ethernet networking profile with a static address to be able to access either of these addresses, and connect directly with an Ethernet cable to the MP device.

For example, to use the MP Fallback IP address, set the PC network profile to:

IP: 172.31.255.253 Netmask: 255.255.255.252 (Note restricted IP and Netmask values)

Alternatively you may set the MP device address to work on your LAN. Connect a telephone to the MP and dial the IVR command **C-O-N-F** (2663) and follow the prompts to set the IP number to one that lies in your normal LAN address range and is not already in use. The device will then reboot. Connect the MP device to an Ethernet port on your LAN and it will be accessible by any PC on the LAN.

From a terminal session on your PC, transfer the required *.img* file to the MP using the *scp* command e.g

```
scp /openwrt-atheros-root-rv287.img root@172.31.255.254:/tmp
```

This will place the file in the */tmp* directory on the MP device. Note that the contents of */tmp* are stored in volatile RAM and thus will be lost on a system restart.

From the *ssh* session install the firmware with the command:

```
sysupgrade -n -v ./<filename>.img
```

The flashing process will begin and may take several minutes, after which the MP device will restart.

Note that the *-n* flag causes previous configuration settings *not* to be retained i.e. the device will operate with the default setting after the flash. This may be an issue for remotely accessed devices – see later section for discussion on this.

After the MP device has restarted and the WiFi led has started to flash, allow up to three minutes for the flashing process to complete. After that you should be able to connect to the MP device with web browser or *ssh* on the default LAN or Fallback IP addresses.

You may also use the IVR **C-O-N-F** (2663) command to change the MP device address to work on your LAN.



## 4.2 Minimum Set-up

The Minimum Set-up process uses the telephone IVR facility to simply set a unique IP address for the **br-lan** bridge interface in order to allow telephone calls to the device using the IP address. Even with this minimal configuration, the MP mesh network may be connected to a LAN and will provide WiFi and Ethernet connectivity for PCs and other devices to the LAN and Internet.

The default setting for the **br-lan** IP address when the device is flashed is 10.130.1.20 and you should change at least the last octet of the address in order to make the address unique on the mesh to support telephone dialling.

In a simple mesh arrangement, all MP devices on the mesh are assigned addresses in the same address range (ie only the last octet of the address is changed) so telephone calls can be made to all devices on the mesh with abbreviated dialling using just the last octet of the MP device's bridge IP address.

If you are intending to connect the mesh to a LAN, you may choose to assign addresses from the LAN address space to the MP devices so that they will appear as static IP devices on the LAN. In this case, just set the IP address of the MP device to the required IP address on the LAN. You will need to ensure that the address that has been assigned will not be used by any other device on the LAN in order to avoid IP conflicts.

### Set the **br-lan** IP Address

Connect a telephone to the MP device and check that you have dial tone.

Use one of the methods below to set the device address.

#### Set Abbreviated Address:

- Pick up the telephone, check for dial tone and dial 2662
- Enter the IVR Pin number (default 1234)
- Follow the voice prompts and enter the required number for the device as 1 – 3 digits e.g 21. This will set the last octet of the MP device IP address e.g. 10.130.1.21
- The number entered will be read back to you and a prompt to reboot the MP.  
The command will fail if the IP is in use (ping) or out of range (.1 to .254).

#### Set Full IP address:

- Pick up the telephone, check for dial tone and dial 2663 (C-O-N-F)
- Follow the voice prompts and enter the IP number in the form 10\*130\*1\*21 (For an IP address of 10.130.1.21)
- The number entered will be read back to you and a prompt to reboot the MP.  
The command will fail if the IP is in use (ping).

After the device has rebooted, you should be able to make a call to the device using either the full IP number, or abbreviated dialling using just the last octet of the address.

**Note:** When the **br-lan** address is set using IVR, the device's **gateway** address will be automatically be set to an address in the same subnet with the last octet set to 1 e.g 10.130.1.1 to ensure correct operation of Asterisk.

If you plan to connect the mesh devices to a LAN and you use this method to set up the MP to have an address in the LAN address space, then the MP will expect to find your LAN router at the **x.y.z.1** address. If your router has a different address, you may use the 4283 (G-A-T-E) IVR command to change the **gateway** address as required after setting the IP address.

### 4.3 Set-up Using SECN Web Interface

#### Basic SECN Configuration

**Mesh Potato SECN Configuration**  
Firmware: SECN Version 1.1-RC4a rv295

[Basic](#) [Advanced](#) [Wireless Status](#)

**Network**

IP Address:  Gateway:  Find Gateway

**WiFi Access Point (WPA default)**

Station ID:  Passphrase:  Channel:

**VoIP / SIP Configuration**

User Name:  Password:

SIP Host:  Dialout Code:

SIP Enable: ☐ SIP Status: **Not Registered**

**Password**

Enter Password:  Repeat Password:

**Web Server Security**

Limit IP Address: ☐ Enable SSL: ☐

The **Basic SECN Configuration** screen may be accessed by pointing your web browser to the IP address of the MP device. A newly flashed device will not have a root password set and thus the web interface will not require authentication.

For a newly flashed device you may use the default IP address of 10.130.1.20 or the Fallback IP address of 172.31.255.253. To use either of these addresses you will need to configure networking on your PC to be able to access these subnets.

Alternatively you may wish to first set the IP address of the MP so that it appears on your LAN subnet by using the phone IVR menu as described in the previous section. If doing so, make sure that you assign an IP address that does not conflict with other devices on the network.

The **Basic SECN Configuration** screen allows you to set up just the key parameters for Network Address, Gateway, WiFi Access point, a SIP/VoIP phone service, set the password for the **root** account, and configure the web server security.

A link is provided at the top of this screen to allow access to the *Advanced SECN* configuration screen if required.

### **Network Configuration**

The network configuration parameters that can be set up are the *IP Address* for the MP device and the IP address for the *Gateway* (router) device on the local network which provides access to the Internet.

The *Find Gateway* button will attempt to locate the Gateway device by sending a DHCP Discover request on the network. If a device responds to the request, then the address of the responding device will be shown in a status message at the bottom on the page. Enter the required Gateway device address in the field and click on the *Save* button.

### **WiFi Access Point Configuration**

The WiFi configuration allows you to set the *Station ID* (SSID), *Passphrase* and radio *Channel* for the MP device.

The *Station ID* must be comprised of alphanumeric characters (plus dash and underscore). This is the name of the WiFi Access-point that will be seen from a WiFi client device attempting to connect.

The *Passphrase* will be required to allow a client to connect if WiFi encryption is being used. The *Passphrase* must be comprised of alphanumeric characters (plus dash and underscore) and must be a minimum of eight characters in length.

Note that as this is the only security that prevents wireless access to your network, you should change the default password and make it suitably strong by using a combination of upper and lower case letters, and numbers.

By default the SECN firmware operates using WPA1-PSK encryption on the WiFi access point. You may change the encryption if required on the *Advanced* screen.

### **VoIP Configuration**

The VoIP configuration allows you to set up a SIP/VoIP service to make calls beyond the local mesh if you wish. Typically this is via a commercial VoIP service provider that provides access to the standard telephone network.

**Note:** *For this to work, the MP device must be correctly configured to operate on a network which provides Internet access for the device through the specified Gateway. In particular, the MP must be configured with an IP address in the same subnet as the Gateway IP address.*

The settings are used to update the `/etc/asterisk/sip.conf` file via the `potato.sip.conf` file.

When you establish an account with a SIP/VoIP provider, you will be given a *User Name* and *Password*, as well as the URL of the SIP server on the Internet. Enter these details in the relevant fields on the screen. The Password will only be displayed when first entered.

The *Dialout Code* is used to distinguish between calls to be made on the local mesh, and those to be routed out to the SIP/VoIP provider. The specified digit needs to be dialled before the required external number. The available digits are #, 0 and 9.

The *SIP Enable* checkbox allows you to enable or disable the external SIP service while keeping the saved settings.

After entering the required settings, click the *Save and Restart Asterisk* button. If the MP can

successfully contact the SIP server and register, the registration status will be shown below the **Dialout** control. Note that registration may take some time and the status may not show immediately. You can click the **Refresh** button to check the status after a period of time.

### **Password**

These fields allow the password to be changed for the **root** account by default. After entering the password in both fields, click on the **Set Password** button to make the change.

A status line at the bottom of the page will indicate whether the change was successful.

### **Web Server**

These controls provide access security configurations to be applied to the web based configuration screens. The options can be applied in any combination and require a restart to become effective.

The **Limit IP Address** checkbox restricts access only to the Fallback IP address 172.31.255.254 with Netmask 255.255.255.252

A connecting PC will need to be set to an IP address of 172.31.255.253 in order to gain access.

The **Enable SSL** checkbox makes access to the unit only available using SSL, thus encrypting data over the link.

When used for the first time, the unit generates a self-signed certificate, which the web browser on a connecting PC will flag and require the user to accept the certificate before allowing access.

When SSL is enabled, the required URL is: ***https://<ip-address>***

### **Saving and Rebooting**


The **Refresh** button will simply refresh the field values from saved values in memory and check the registration status of the SIP/VoIP service.

The **Save** button writes the various field values on the screen into memory, but does not change the operation of the MP device.

The **Save and Restart Asterisk** button will save all field values and restart the Asterisk telephony subsystem. This usually takes about 10 seconds to complete, but may take longer if Internet access is not configured correctly.

The **Save and Reboot** button will save the field values and restart the MP device using the newly saved values. Note that **a restart is required** to effect changes to the Network, WiFi and Web Server security settings, and will take around two minutes to complete.

## Advanced SECN Configuration



**VillageTelco**

**Mesh Potato SECN Configuration**

Firmware: SECN Version 1.1-RC4a rv295

Basic
Advanced
Wireless Status

**Network**

IP Address	<input type="text" value="10.130.1.20"/>		Gateway	<input type="text" value="10.130.1.1"/>
DNS	<input type="text" value="8.8.8.8"/>		Netmask	<input type="text" value="255.255.255.0"/>

**WiFi Access Point**

SSID	<input type="text" value="VT-SECN-AP"/>	US/Can (11 ch)	<input type="checkbox"/>	Channel	<input type="text" value="1"/>
Passphrase	<input type="text" value="potato-potato"/>	Encryption	<input type="text" value="WPA1"/>	AP Enable	<input type="text" value="Enabled"/>

**Mesh Wireless Interface**

IP Address	<input type="text" value="10.10.1.20"/>		Netmask	<input type="text" value="255.255.255.0"/>
SSID	<input type="text" value="vt-mesh"/>		BSSID	<input type="text" value="02:CA:FF:EE:BA:BE"/>
Tx Power (10 - 20)	<input type="text" value="17"/>		Encryption	<input type="text" value="OFF"/>
MP Gateway Mode	<input type="text" value="OFF"/>		Wifi Mode	<input type="text" value="802.11G"/>

**VoIP / SIP Configuration**

SIP Registrar	<input type="text" value="sip.myhost.com"/>	User Name	<input type="text" value="myusername"/>
SIP Host	<input type="text" value="sip.myhost.com"/>	Password	<input type="text"/>
Enable Asterisk NAT	<input type="checkbox"/>	NAT External IP	<input type="text" value="0.0.0.0"/>
SIP Enable	<input type="checkbox"/>	Register	<input type="checkbox"/>
		Dialout Code	<input type="text" value="#"/>
SIP Status <span style="color: green;">Not Registered</span>			
Softphone Support	<input type="text" value="CLIENT"/>		
Codec1	<input type="text" value="gsm"/>	Codec2	<input type="text" value="ulaw"/>
		Codec3	<input type="text" value="alaw"/>

**DHCP Server**

Starting IP	<input type="text" value="10.130.1.200"/>		Ending IP	<input type="text" value="10.130.1.240"/>
Lease Term (secs)	<input type="text" value="7200"/>		Max Leases	<input type="text" value="40"/>
DNS	<input type="text" value="8.8.8.8"/>		Domain Name	<input type="text" value="lan"/>
Subnet Mask	<input type="text" value="255.255.255.0"/>		Router	<input type="text" value="192.168.1.1"/>
Enable DHCP Server	<input type="checkbox"/>			

The **Advanced SECN Configuration** screen may be accessed by clicking on the link at the top of the **Basic SECN Configuration** page.

This screen allows you to set up basic and additional parameters for Network, WiFi, a SIP/VoIP phone service, Softphone support and DHCP server.

Links are provided at the top of this screen to allow access to the **Basic SECN** and **Wireless Status** configuration screens if required.

### **Network Configuration**

The network configuration parameters that can be set up are the **IP Address** and **Netmask** for the MP device, the IP address for the **Gateway** (router) device on the local network, which provides access to the Internet, and the IP address of the **DNS** server to be used for name resolution.

### **WiFi Access Point Configuration**

The WiFi configuration allows you to set the **Station ID** (SSID), **Passphrase**, **Encryption** and radio **Channel**, and the maximum number of connections for the MP device.

The **US/Can (11ch)** checkbox sets the regulatory domain for North America to limit the number of available channels to 11 in accordance with FCC regulations. When this mode is active and channel 12 or 13 is selected, the channel setting will be set to Channel 1.

The **Station ID** must be comprised of alphanumeric characters, plus dash and underscore. This is the name of the WiFi Access-point that will be seen from a client device attempting to connect.

The **Passphrase** will be required to allow a client to connect if WiFi encryption is being used. The passphrase must be comprised of alphanumeric characters (plus dash and underscore) and must be a minimum of eight characters in length.

Note that as this is the only security that controls access to your network, you should change the default password and make it suitably strong by using a combination of upper and lower case letters, and numbers.

The **Encryption** control allows you to select from WPA1, WPA2, WEP or Open encryption on the WiFi Access-point.

The **AP Connections** control sets the maximum number of WiFi associations that will be supported. This may be used to manage the load on APs in an extended WiFi cell arrangement, or to disable the AP.

### **Mesh Wireless Interface Configuration**

The Mesh Wireless configuration allows you to set a number of parameters for the mesh **ath0** interface including: IP Address, Netmask, SSID, BSSID, Transmit Power, and Country Code. These values are written into the configuration files **/etc/config/network** and **/etc/config/wireless**.

The **ath0** interface used for the mesh wireless protocol has an **IP Address** and **Netmask**. These are set to default values of 10.10.1.20 and 255.255.255.0 and normally do not need to be altered. These settings are **not** used for the OSI Layer 2 Batman-adv mesh protocol, and so all MP devices on the mesh may remain on the default IP address.

The **ath0** IP address can be used to access the MP device for maintenance in the same way as the Network Address or the Fallback Address described previously. If it is intended to use this address for maintenance access, it should be set to a unique value to avoid any potential IP address conflict.

The **SSID and BSSID** parameters set the station identification for the MP on the mesh and should be set the same for all devices in a mesh cell. These parameters can be used to set up separate mesh cells if required.

**Note:** It is a requirement of the current OpenWrt operating system that the **BSSID** must commence with an even number eg 02, 04, 06 etc.

The **Tx Power** parameter may be used to adjust the power of the device radio transmitter. It is set by default to the maximum value of 17. Normally this should not need to be adjusted but doing so may be useful in certain circumstances such as testing.

The **Encryption** control may be used to enable encryption on the mesh, if the device supports it.

The **Gateway Mode** determines whether the device will act as a Gateway in the Batman-adv mesh routing protocol. This setting is only needed if there is **more than one gateway** device on the mesh. A device which is connected to a LAN in order to provide Internet access for example, should be set to **Server** mode to assist routing requests efficiently through the mesh. Devices requiring access through a Gateway should have this set to **Client** mode.

The **WiFi Mode** control allows selection of the hardware modes supported by the device eg 802.11G and 802.11N-G.

### VoIP / SIP Configuration

The VoIP / SIP configuration allows you to set up a SIP/VoIP service to make calls beyond the local mesh if you wish, and to optionally support Softphone operation on devices attached to the mesh.

Typically VoIP / SIP operation is via a commercial VoIP service provider that provides access to the standard telephone network.

**Note:** *For this to work, the MP device must be correctly configured to operate on a network which provides Internet access for the device through the specified Gateway. In particular, the MP must be configured with an IP address in the same subnet as the Gateway IP address.*

The settings are used to update the `/etc/asterisk/sip.conf` file via the included `potato.sip.conf` file.

When you establish an account with a SIP/VoIP provider, you will be given a **User Name** and **Password**, as well as the URL of the SIP Host and Registrar server on the Internet. Enter these details in the relevant fields on the screen.

The **Enable Asterisk Nat** checkbox may be used to enable Asterisk use behind a NAT firewall. Normally this is not required for a LAN behind a simple router/NAT firewall providing Internet access, but may be required, for example, if the MP is behind a second NAT firewall. If used, the **External NAT IP** field should be set to the **upstream** network IP address of the NAT router to which the MP is connected.

The **Dialout Code** is used to distinguish between calls to be made on the local mesh, and those to be routed out to the SIP/VoIP provider. The specified digit is required to be dialled before the required external number. The available digits are #, 0 and 9.

The **SIP Enable** checkbox allows you to enable or disable the external SIP service while keeping the saved settings.

The **Register** checkbox determines whether the device will register with the SIP host. Registration is required in order to receive incoming calls, and some providers require registration for outgoing calls as well.

The **Softphone Support** control is used to set the mode of operation of the MP device in conjunction with other devices such as cell phones or laptops attached to the network typically via WiFi. See later section for details of Softphone operation.

**NOTE: One device only** on the mesh may be set to **Master** mode, and this device will automatically be configured to use the reserved IP address of .252 on the LAN segment in use.

The **Codec** settings may be used to control the Codecs available to be used for calls. Normally this will not need to be changed, however some SIP/VoIP providers do require specific codecs to be used, in particular for calls outside their immediate domain.

After entering the required settings, you may click the **Save and Restart Asterisk** button for the changes to be made effective. If the MP can successfully contact the SIP/VoIP server and register, the registration status will be shown alongside the **Sip Status** label.

Note that registration may take some time and the registered status may not show up when the screen is first refreshed. You can click the **Refresh** button to check the status.

### DHCP Server Configuration

The DHCP configuration allows you to set up a DHCP server to operate on the device. This may be used, for example, to ensure that devices attaching to the mesh network are able to obtain an IP address via DHCP in the event that there is no service available from a gateway device, perhaps due to the absence of an uplink to a remote router device.

**Note:** Care must be taken in setting up the configuration of the DHCP server to ensure that there is no conflict between multiple DHCP servers that are visible to devices attached to the network. Normally only a single DHCP server is enabled on a network.

The DHCP server provides IP address leases and a range of network information to clients in response to a DHCP Discovery request. The settings for the DHCP server that can be configured from the MP device web interface are outlined below.

The **Enable DHCP Server** checkbox allows the server in the MP device to operate when it is checked. By default the DHCP server is not enabled.

The **Starting** and **Ending IP** fields set the range of addresses that will be handed out by the DHCP server. Care must be taken to ensure that this range does not overlap the range of any other DHCP server on the network.

**Lease Term** sets the time period in seconds that IP address leases are valid.

**Max Leases** sets the maximum number of concurrent leases that will be handed out.

**DNS** defines the Domain Name Server IP address that will be handed out to clients as part of the DHCP protocol.

**Domain Name** sets the network name that will be handed out to clients as part of the DHCP protocol.

**Subnet Mask** sets the network mask that will be handed out to clients as part of the DHCP protocol.

**Router** sets the IP address of the network gateway that will be handed out to clients as part of the DHCP protocol.



### Saving and Rebooting

The ***Refresh*** button will simply refresh the field values from saved values in memory and check the registration status of the SIP/VoIP service.

The ***Save*** button writes the various field values on the screen into memory, but does not change the operation of the MP device.

The ***Save and Restart Asterisk*** button will save all field values and restart the Asterisk telephony subsystem. This usually takes about 10 seconds to complete, but may take longer if Internet access is not configured correctly.

The ***Restore Defaults*** button will reset all the configuration settings to the default values of a newly flashed device.

The ***Save and Reboot*** button will save the field values and restart the MP device using the newly saved values. Note that ***a system restart is required*** to effect changes to the network settings and will take around two minutes to complete.

## 4.4 Advanced Set-up

Advanced Set-up utilises access to the Linux operating system on the device and permits full access to all configuration facilities as well as adding and configuring additional software packages.

Access to the device for Advanced Set-up is by means of a command line from a *ssh* terminal session.

### Connecting to the device

When first flashed with new SECN firmware, the device supports a *telnet* connection as there is no *root* password set.

Connect to the MP with *telnet* using the MP device's Fallback address: 172.31.255.254

Set PC network to: IP: 172.31.255.253 Netmask: 255.255.255.252

Alternatively the default *br-lan* IP address 10.130.1.20 may be used with Netmask 255.255.255.0

Once the telnet connection has been made, set the *root* password with the *passwd* command, logout with the *exit* command, then reconnect with *ssh*.

### Setting the device Network Addresses

#### Setting the *br-lan* Bridge IP Address

Set the unique IP address for the *br-lan* interface of the MP device by using the *uci* command, or by directly editing the network configuration file.

From the command line:

```
uci set network.br-lan.ipaddr=103.130.1.xxx  (Where xxx is unique to each MP)
uci commit network
```

Editing the */etc/config/network* file:

```
config 'interface' 'lan'
    option 'type' 'bridge'
    option 'ifname' 'eth0 bat0 ath1'
    option 'proto' 'static'
    option 'netmask' '255.255.255.0'
    option 'gateway' '10.130.1.1'           # Default router address
    option 'dns' '8.8.8.8'
    option 'ipaddr' '10.130.1.xxx'         # Where xxx is unique to each device
```

### Setting the *ath0* IP Address

You may wish to change the *ath0* IP address, however this is not required for basic mesh operation.

From the command line:

```
uci set network.wifi0.ipaddr=10.130.1.xxx    (Where xxx is unique to each MP)
uci commit network
```

Editing the */etc/config/network* file:

```
config 'interface' 'wifi0'
    option 'ifname' 'ath0'
    option 'proto' 'static'
    option 'ipaddr' '10.10.1.xxx'
    option 'netmask' '255.255.255.0'
    option 'mtu' '1527'
```

### Set the Access Point SSID and WPA Passphrase

From the command line:

```
uci set secn.accesspoint.ssid= VT-SECN-AP
uci set secn.accesspoint.passphrase = potato-potato
uci commit secn
```

Editing the */etc/config/secn* file: (MP file example)

```
config 'mesh' 'accesspoint'
    option 'wpa_key_mgmt' 'WPA-PSK'
    option 'encryption' 'WPA1'
    option 'ssid' 'VT-SECN-AP'
    option 'passphrase' 'potato-potato'
    option 'ap_enable' '1'
```

NOTE: On the MP device running SECN-1.1 firmware, the *secn* config file parameters are used to automatically generate the *hostapd* configuration file. Do not edit the *hostapd* configuration file as it will be overwritten on startup or on use of the web interface.

## Modifying Asterisk Operation

### Setting up External SIP / VoIP Operation

To add external VoIP support, use the SECN web configuration interface or modify the *secn* configuration file.

From the command line:

```
uci set secn.asterisk.host = sip.myhost.com
uci set secn.asterisk.reghost = sip.myhost.com
uci set secn.asterisk.fromdomain = sip.myhost.com
uci set secn.asterisk.secret = mysecret
uci set secn.asterisk.username = myusername
uci set secn.asterisk.fromusername = myusername
uci commit secn
```

Editing the */etc/config/secn* file:

```
config 'mesh' 'asterisk'
    option 'fromdomain' 'sip.myhost.com'
    option 'host' 'sip.myhost.com'
    option 'reghost' 'sip.myhost.com'
    option 'secret' 'mysecret'
    option 'username' 'myusername'
    option 'fromusername' 'myusername'
    option 'codec1' 'gsm'
    option 'codec2' 'ulaw'
    option 'codec3' 'alaw'
    option 'enablenat' ''
    option 'externip' '0.0.0.0'
    option 'proxy' ''
    option 'softph' 'CLIENT'
    option 'dialout' '#'
    option 'enable' 'checked'
    option 'register' 'checked'
```

### Dial Plan for SIP / VoIP

The dial plan for external SIP / VoIP operation is defined in the configuration include file */etc/asterisk/potato.extensions.conf* as follows:

```
; Send incoming calls to the MP
exten => s,1,Dial(MP/1)
; Make outgoing calls using [sipaccount] details
; Dial # for access, and then required number string
exten => _#,1,Dial(SIP/${EXTEN:1}@sipaccount,120,r)
```

## 5. Overview of SECN-1 Operation

This configuration uses Batman-advanced for the mesh rather than Batman as used in earlier firmware versions. Batman-advanced uses a different mesh protocol to batman and so the two will not interoperate on the same mesh.

The MP device provides two physical network interfaces, Ethernet cable and wireless, which are configured as follows:

- The **eth0** interface operates on the MP Ethernet cable connection.
- Two wireless interfaces, **ath0** and **ath1**, are set up on the wireless interface **wifi0**.
- Batman-adv is configured to run on the **ath0** interface using the **batctl** command and generates the **bat0** interface.
- The second wireless interface, **ath1**, is set up to operate as a WiFi access point using the **iwconfig** command.
- The **bat0**, **ath1** and **eth0** interfaces are bridged (**br-lan**) together in each MP and assigned a static IP address, and thus, due to the operation of the mesh via **bat0**, all the **ath1** and **eth0** interfaces of all the MPs in the mesh are similarly bridged.
- The default IP address range used for the **br-lan** interface is 10.130.1.xxx

The mesh will operate in a stand-alone configuration, simply connecting attached devices together and providing telephony between devices. Alternatively the mesh may be interconnected to a LAN to provide access to additional resources, including Internet connectivity.

If one of the devices is connected via Ethernet cable to a LAN router, then all WiFi and Ethernet interfaces connected to the meshed devices will have access to the LAN resources.

If there is a DHCP server running on the LAN (eg in the router/gateway) then devices configured as DHCP clients connected to the MPs via WiFi or Ethernet will acquire an IP address just as if they were connected directly to the LAN.

Note that there is no DHCP server running in a stand-alone mesh arrangement by default, and so in this case, attached devices would need to be statically configured for their IP address in order to connect. Alternatively one of the meshed devices may be configured to provide DHCP service.

### 5.1 IP Address Range for MPs

It should be noted that the IP address used for the **br-lan** bridge in the MP devices needs to be configured during setup, and **may** or **may not** be made to lie in the IP address space used on the LAN to which the mesh may be connected. Operation is essentially the same in both cases, but care must be taken to manage the address space in the former case to avoid conflicts with LAN addresses.

IP addresses assigned to MP devices are static. If the IP addresses used for the MP devices lie in the same address space as the LAN, then the DHCP server and other devices on the LAN must be appropriately configured so that the addresses assigned to the MP devices are left free in order to avoid IP address conflicts. In this arrangement, the MP devices will appear on the LAN just as any other device with a static IP address, and they may be accessed for management via browser or ssh terminal session.

Conversely, if the IP address range used for the MP devices is separate to that used on the LAN, the

MP devices will not appear on the LAN and there is no need to reserve the address space. In order to access the MPs for management in this configuration, it is necessary to configure a PC with a static address in the same range as the MPs, and attach via Ethernet cable or WiFi.

The default IP address assigned to the **br-lan** interface in the MPs is 10.130.1.20 which is unlikely to conflict with the default address range of commodity routers.

If it is desired to have the MPs appear on the LAN, the **br-lan** IP address should be assigned accordingly during set up.

The address assigned to the **br-lan** interface for each MP must be changed to be unique, so that each device can provide a separate telephone number. This IP address assignment may be made by a number of methods including telephone IVR, web interface or manipulation of the `/etc/config/network` file.

## 5.2 Batman-Adv Operation

Batman-adv is a "OSI layer 2" routing protocol which is implemented as a kernel module in the Linux kernel. Since Linux 2.6.38 batman-adv is an official part of Linux.

When you assign at least one active physical network interface to batman-advanced, it will create the virtual **bat0** interface. In the SECN-1 firmware **ath0** is assigned to the batman-advanced kernel module. **ath0** is the wireless interface operating in multipoint-to-multipoint mode (ad-hoc).

Because batman-adv operates entirely on MAC layer (OSI layer 2), **ath0** doesn't need any Layer 3 configuration. Only its Layer 2 MAC address is required. The MAC address is configured during production, so we don't need to configure it. All we need to do is make sure to switch **ath0** on. To sum it up: **ath0** is the link-local transport interface for the batman-advanced mesh.

Batman-adv itself bridges all **bat0** interfaces in all the mesh devices to a big, smart, virtual switch. This means that all **bat0** interfaces in the mesh are link-local - even if they are multiple wireless hops away.

Despite being virtual, **bat0** acts like a real, physical, network interface connected to a big switch. As such you can run all kinds of network protocols on it, like IPv4, IPv6, ARP, Zeroconf (yes, you can run mDNS on **bat0**!), IPX – or whatever protocol that can communicate over a network interface that is connected link local (which means directly connected, like a straight Ethernet cable connected between two computers, or a bunch of computers connected to a switch).

In the SECN firmware the **bat0** interface itself is again assigned (or rather enslaved) to a bridge in each machine. **bat0** is part of the bridge named **br-lan**, together with **ath1** and **eth0**.

**eth0** is the LAN port of the MP. **ath1** is a access-point interface, operating as a master in WiFi infrastructure mode. (As opposed to a infrastructure client, like laptops or smartphones with a WiFi interface).

Hence **all eth0** and **ath1** interfaces in **all** devices running the SECN firmware are part of **one** big wireless bridge. The **ath0** interface does the low level work to carry the traffic link-locally from hop to hop and batman-advanced takes care about the routes that the MAC packets have to take.

**Note:** It is not possible to add IP settings to an interface which is encapsulated in a bridge - you can only assign IP settings to the bridge interface itself. **eth0** is part of the bridge **br-lan**, together with **ath1**, **bat0** (the batman-adv virtual interface, which is routed by the mesh routing protocol using MAC addresses). Hence you can not assign any IP settings to **eth0**, **ath1** or **bat0** - only to **br-lan**.

## BATCTL Command

The following description is taken from the man page published by OpenMesh.org at:  
<http://downloads.open-mesh.org/batman/manpages/batctl.8.html>

**Syntax**        **batctl** [*batctl-options*] *command* [*command-options*]

This command offers a convenient way to configure the batman-adv kernel mod-ule as well as displaying debug information such as originator tables, translation tables and the debug log. In combination with a bat-hosts file batctl allows the use of host names instead of MAC addresses.

B.A.T.M.A.N. advanced operates on layer 2. Thus all hosts participating in the virtual switched network are transparently connected together for all protocols above Layer 2. Therefore the common diagnosis tools do not work as expected. To overcome these problems batctl contains the commands **ping**, **tracertoute**, **tcpdump** which provide similar functionality to the normal **ping(1)**, **tracertoute(1)**, **tcpdump(1)** commands, but modified to Layer 2 behaviour or using the B.A.T.M.A.N. advanced protocol.

Commands of particular interest include the following:

**originators|o**    [-w [*interval*]][-n][-t]

Once started batctl will display the list of announced gateways in the network. Use the "-w" option to let batctl refresh the list every second or add a number to let it refresh at a custom interval in seconds (with optional decimal places). If "-n" is given batctl will not replace the MAC addresses with bat-host names in the output. The "-t" option filters all originators that have not been seen for the specified amount of seconds (with optional decimal places) from the output.

**gw\_mode|gw**      [*off|client|server*] [*sel\_class|bandwidth*]

If no parameter is given the current gateway mode is displayed otherwise the parameter is used to set the gateway mode. The second (optional) argument specifies the selection class (if 'client' was the first argument) or the gateway bandwidth (if 'server' was the first argument). If the node is a server, this parameter is used to inform other nodes in the network about this node's internet connection bandwidth. Just enter any number (optionally followed by "kbit" or "mbit") and the batman-adv module will guess your appropriate gateway class. Use "/" to separate the down- and upload rates. You can omit the upload rate and the module will assume an upload of download / 5.

default: 2000 -> gateway class 20

examples: 5000 -> gateway class 49

5000kbit

5mbit

5mbit/1024

5mbit/1024kbit

5mbit/1mbit

If the node is a gateway client the parameter will decide which criteria to consider when the batman-adv module has to choose between different internet connections announced by the aforementioned servers.

### **bat-hosts file**

This file is similar to the */etc/hosts file*. You can write one MAC address and one host name per line. **batctl** will search for **bat-hosts** in */etc*, your **home** directory, and the **current** directory. The found data is used to match MAC address to your provided host name or replace MAC addresses in debug output and logs. Host names are much easier to remember than MAC addresses.

## Batman-adv and Gateways

Amongst performance improvements and faster handover of clients, the batman-adv package for the MP now supports configuring advanced batman-adv gateway and gateway client parameters via UCI.

**Note:** You only need this if you want to use **more than one gateway** in the mesh. In this case set the gateway MPs mode to Server and the other MPs mode to Client

Gateway settings for **Server** and **Client** mode are provided in the SECN web interface on the Advanced page.

Settings may be made from the command line as follows.

An example how to configure batman-adv gateway bandwidth:

```
root@MP-2:/# uci set batman-adv.bat0.gw_mode=server
root@MP-2:/# uci set batman-adv.bat0.gw_bandwidth=384kbit/128kbit
root@MP-2:/# uci commit
root@MP-2:/# /etc/init.d/batman-adv restart
```

Setting the downlink/uplink speed of the gateway like in this example is optional, if you want to override the default value.

For more info check out <http://downloads.open-mesh.org/batman/manpages/batctl.8.html>

An example how to configure a MP as batman-adv gateway client:

```
root@MP-2:/# uci set batman-adv.bat0.gw_mode=client
root@MP-2:/# uci commit
root@MP-2:/# /etc/init.d/batman-adv restart
```

By default the clients will connect to the gateway with the 'best' access (qualified by the TQ route metric, which measures route quality) in the mesh. As long as no other gateway has a TQ route metric which is more than 20 counts better than the currently selected gateway, the clients will stick to the current gateway. If another gateway is more than 20 TQ counts better, the clients will switch the selected gateway. You can, of course, tweak this threshold.

If you want to do some fancy - experimental - setup like dynamically changing the announced gateway bandwidth, in order to balance the load of the gateways, you can change the clients gateway selection algorithm.

Example:

```
uci set batman-adv.bat0.gw_mode=client 1
uci commit
/etc/init.d/batman-adv restart
```

This setting will configure the clients to select the gateway in order to find the best compromise of TQ route metric and announced gateway speed. A 100Mbit/100Mbit gateway is useless if the TQ route metric is small.

For more detailed info check out the batctl man page at [open-mesh.net](http://open-mesh.net)



## 5.3 Telephony Operation

### Overview

MP devices provide an RJ11 port to which a telephone may be connected and each MP device runs a copy of the Asterisk application to provide the telephony facilities. Asterisk allows phone calls to be made between devices by means of Voice over IP (VoIP) and Session Initiated Protocol (SIP).

### Interactive Voice Response (IVR) Commands

The MP Asterisk configuration includes several telephone extension numbers that allow interaction with the device using Interactive Voice Response (IVR) system. These numbers include:

2661	Read out the <b>ath0</b> mesh wireless interface IP address
2664	Read out the <b>br-lan, eth0, ath1</b> network interface IP address
7774 RSSI	Read out the <b>rss</b> i signal strength
2426 CHAN	Set wireless channel
2662	Set the unique <b>network</b> IP address of the MP device – <b>Last octet</b> .
2663 CONF	Set the unique <b>network</b> IP address of the MP device – <b>Full IP</b> .
4283 GATE	Set the IP address of the network gateway used by the MP device.
6749 MPGW	Set the mesh batman-adv gateway mode of the MP device.
7466 PINN	Set IVR PIN number. Default pin is <b>1234</b>
9434 WIFI	Set WiFi passphrase.
3427 DHCP	Enable DHCP temporarily on br-lan to offer Fallback IP.
9322 WEBB	Enable / Disable the web interface
73738 RESET	Restore factory default configuration settings.
9999	Restart Asterisk.

### IVR Command Summary

Commands which change the system configuration require the entry of the IVR PIN number for security. The initial IVR PIN number is set to **1234** This should be changed before deployment.

Commands which have a variable number of input digits will wait for a timeout period after the last digit has been input to complete the command. The command may be terminated immediately by keying in the **#** digit after the last command digit has been entered.

Commands which require the MP device to be restarted to take effect (e.g. network settings) will include a message advising this fact.

The **2661** and **2662** commands simply read out the IP addresses used by the MP device on the mesh and on the wifi and ethernet network carried on the mesh, respectively.

The **RSSI** command **7774** will read out the signal strength of neighbouring MP devices. This is intended to assist with adjusting the MP device's location and orientation for best signal from its neighbours on the mesh.

If the mesh has **batman-adv** gateways set up the RSSI command will list the signal strength values for each neighbour which is selected as a next hop towards a gateway as follows:

**Gateway nexthop ... 1 ... 34, Gateway nexthop ... 2 ... 22,**

In the absence of gateways the RSSI command will read out the signal strength values for all neighbours as follows:

**Neighbour 1.....36, Neighbour 2.....42, Neighbour 3.....28**

Other useful terminal commands for monitoring signal strength are:

**wlanconfig ath0 list** Lists signal data for nearby devices on the mesh.

**wlanconfig ath1 list** Lists signal data for devices attached to the MP's WiFi Access Point.

**batctl o** Lists nearby devices on the mesh.

**batctl gw server** Enable batman-adv gateways on the fly

The **CHAN** command **2426** sets the wireless channel used for the mesh and wifi interfaces.

The **CONF** command **2663** is used to set the unique network address of the MP device using the full IP number. The **2662** command changes only the last octet of the IP address. If the MP device is attached to a network and the specified IP address is already in use, the commands will fail.

The **GATE** command **4283** sets the IP address of the network gateway that the MP can use to access IP addresses beyond the local network, such as Internet addresses.

The **MPGW** command **6749** sets the **batman-adv** gateway mode of the MP. This setting is used to assist with efficient routing of traffic on the mesh. If more than one MP on the mesh is connected to a gateway, these MP devices should have their gateway mode set to **Server**, with other MP devices set to **Client**. If only one MP device on the mesh is connected to a gateway then it is useful to announce this device by setting its gateway mode to **Server**.

The **PINN** command **7466** sets the four digit IVR PIN number, which should be changed from the default 1234 before the device is deployed in the field.

The **WIFI** command **9434** sets the encryption passphrase used for secure wifi access as a numeric string. A minimum of eight characters is required and the passphrase should be changed from the default before field deployment.

The **DHCP** command **3427** activates a DHCP server temporarily on the device so that if you connect a PC via Ethernet or WiFi it will be automatically given an IP address corresponding to the MP Fallback address. This avoids having to set up a static IP on the PC to connect. The facility can be activated and de-activated with this command, and it is deactivated automatically on a reboot.

The **RESET** command **73738** restores the device to the original factory default settings.

The **Restart Asterisk** command **9999** can be useful to ensure that the telephony sub-system is initiated correctly after the mesh network starts up and Internet access becomes available for registering external SIP providers. Restart time for Asterisk is a few seconds, after which dial tone will be available.

## 5.4 Asterisk Operation

The operation of Asterisk is controlled to a number of configuration files, two of which are of particular interest for MP devices - */etc/asterisk/extensions.conf* and */etc/asterisk/sip.conf*. The *extensions.conf* file sets up the dial plan while the *sip.conf* file defines the channels to be used for making calls.

Operation of Asterisk can be monitored from the MP command line by executing the commands:

```
# asterisk -r
```

```
# asterisk -vvvvvrddd
```

Launches with Verbose Lev 5 and Core Debug Lev 3.

Some useful commands in the Asterisk shell include:

CLI> exit	Return to the command shell
CLI> help	Displays a list of available commands
CLI> core set verbose 5	Set verbose level to 5
CLI> sip reload	Reload sip.conf configuration
CLI> dialplan reload	Reload extensions.conf dialplan
CLI> show dialplan default	Display current dial plan
CLI> sip show registry	Display sip registrations

## Making Calls to MP Devices

To dial an MP device using the full IP address, dial the IP number substituting the \* character for the dots between octets in the address. To dial an MP with address 10.130.1.21, dial

10\*130\*1\*21

The SECN firmware includes a facility for making on mesh calls using abbreviated dialling by using just the last octet of the MP device's IP address. When an abbreviated number dial string is detected, the full IP address is generated by pre-pending the rest of the address.

The IP address used for on mesh abbreviated dialling is set up during the start up process by the script */bin/generate-extension.sh*, using the MP device's own *br-lan* IP address as reference.

To dial an MP device using abbreviated dialling, simply dial the last octet of the unique IP number assigned to the required MP. This can be dialled as 1, 2 or 3 digits, and may include leading 0 eg

5, 05, 005	(device address 10.130.1.5)
25, 025	(device address 10.130.1.25)
105	(device address 10.130.1.105)

## Debugging Asterisk Operation

Asterisk provides a comprehensive interactive console mode to allow you to monitor its operation.

If Asterisk is already running, invoke the console mode with the command:

```
# asterisk -vvvvvrddd
```

This will run the console with Verbosity set to Level 5, and Core Debug set to level 3, which generally gives good visibility of what is happening. It does not interfere with the operation of Asterisk.

An extensive set of commands is available from the Asterisk CLI command line.

To see a list of these commands type: **help**

To exit the console mode, type: **exit**

If Asterisk is not already running, you can start it up with console mode running with the command:

```
# asterisk -vvvvvrgcddd
```

This can be useful for monitoring the start-up behaviour of Asterisk.

## Asterisk and Network Settings

Asterisk has some very particular requirements around network settings, specifically:

### Network DNS Address

Firstly, during Asterisk start-up, it will test for the presence of a ping response from the DNS **nameserver** address specified in the **/etc/resolv.conf** file. It may wait for a period of many seconds for a response, which will affect the start up delay for the whole device. This delay can be seen in the Asterisk console. In the MP device firmware, the Asterisk start-up script temporarily uses the local host address for the DNS setting to ensure fast start-up.

Secondly, for external SIP / VoIP operation, Asterisk will use the **nameserver** value in **/etc/resolv.conf** to resolve the URL of the SIP / VoIP host server on the internet. If there is no valid DNS service operating on this address, or the DNS address is not accessible from the MP device, Asterisk will fail to register the SIP / VoIP service and will complain of a **DNS error** in the Asterisk interactive console output.

### Network Gateway Address

Asterisk requires that the network **gateway** address specified in **/etc/config/network** be in the same IP subnet range as the MP's IP address, **even if there is no device actually present at this address**.

If the **gateway** address is not in the correct subnet, Asterisk will fail to place even on-mesh calls and will complain of a '**Bad file descriptor**' error in the interactive console output.

When the MP device's unique IP address is set from the IVR, the Gateway addresses will be set to be in the same IP subnet with the final octet set to '1' e.g. **10.130.1.1**

By default, the IVR function will set the DNS address to a public server address at **8.8.8.8**

**Note: Care must be taken when setting these addresses manually from SECN web interface or command line.**

### Access to SIP/VoIP Server

If Asterisk cannot access the network and see the external VoIP host during startup, calls through the service will fail, even if Asterisk is able to register with the service after startup. Calls to mesh devices will work correctly in this scenario, leading to confusion over the status of Asterisk.

This is particularly relevant to MP devices that are connected to the LAN / Internet only via the mesh, as the start up order and timing of scripts in **/etc/rc.d** are designed to ensure the mesh is running correctly before Asterisk tries to start.

## **Sample Asterisk Console Outputs**

### **1. Call from MP at 192.168.1.32 to MP at 192.168.1.22 on the mesh network.**

```
MP-32*CLI>
-- event_offhook
-- AST_STATE_DOWN:
-- start mp_new
-- event_dtmf 2
-- event_dtmf 2
-- event_digit_timer
-- extension exists, starting PBX 22
-- Executing [22@default:1] Dial("MP/1", "SIP/4000@192.168.1.22") in new stack
-- Called 4000@192.168.1.22
-- SIP/192.168.1.22-00587578 is ringing
-- Asked to indicate 'Remote end is ringing' condition on channel MP/1
MP-32*CLI>
```

### **2. Call to a PSTN number 0733991234 via SIP / VoIP Service**

```
MP-32*CLI>
-- event_offhook
-- AST_STATE_DOWN:
-- start mp_new
-- event_dtmf #
-- event_dtmf 0
-- event_dtmf 7
-- event_dtmf 3
-- event_dtmf 3
-- event_dtmf 9
-- event_dtmf 9
-- event_dtmf 1
-- event_dtmf 2
-- event_dtmf 3
-- event_dtmf 4
-- event_digit_timer
-- extension exists, starting PBX #0733991234
-- Executing [#0733991234@default:1] Dial("MP/1", "SIP/0733991234@sipaccount|120|r")
-- Called 0733991234@sipaccount
-- Asked to indicate 'Remote end is ringing' condition on channel MP/1
MP-32*CLI>
```

## 5.5 Softphone Support

Softphone Support is provided in order to be able to allow devices such as cell phones and laptop PCs equipped with softphone applications to join the MP telephone network and to make and receive calls on the network, and to an external SIP/VoIP service if configured.

### Setting up the Devices

Softphone Support is enabled by the control in the VoIP / SIP section of the *Advanced SECN Configuration* screen. The available modes are *Off* (default), *Master* and *Client*.

In order to support Softphones on a network over the mesh, *one, and one only*, device on the network is set to *Master* mode. The copy of Asterisk running on the Master device is used to route softphone calls around the network.

The *Master* device will automatically have its IP address last octet set to *.252*

This address is reserved by default in a SECN network as a 'well known' network address for the Softphone server.

Other MP devices on the network that are to be able to make calls to softphone equipped devices must have their Softphone Support control set to *Client*.

Note that after setting the mode in the configuration screen, the device has to be restarted for the changes to take effect.

### Configuration of Softphone Accounts

Softphone accounts are defined in the file */etc/asterisk/softphone.sip.conf*

By default there are ten accounts set up for softphones defined as *softph300* through *softph309*

Once assigned to particular attached softphone devices, these devices may be called using their three digit numbers 300 through 309.

The list of softphone accounts may be extended as required, and the individual passwords changed as required by manually editing the configuration file.

Note that setting of the allowed codec(s) is critical to the operation of some softphone clients. It has been found for example that SipDroid will operate correctly only when *ulaw* is the only allowed codec.

A section of the *etc/asterisk/softphone.sip.conf* file is shown below for reference.

```
[softph300]
type=friend
secret=Pa55uu0rd300
context=default
host=dynamic
disallow=all
;allow=gsm
allow=ulaw
;allow=alaw
dtmfmode=rfc2833
qualify=yes
canreinvite=no
nat=yes
```

### **Setting up the DHCP Server**

Telephony on the SECN-1 MP network relies on the IP addresses of the devices attached to the network to route calls to the correct device. MP devices typically have statically assigned IP addresses for this purpose. This allows a MP network to operate without the need for any master device controlling the telephony system, thus providing maximum robustness.

This is not the case with Softphone Support described here. Softphone devices are 'registered' with the Softphone Master device, and the presence and correct operation of this device is essential for softphone operation. It is a single point of failure.

Furthermore, the softphones do not rely on their IP address to determine their phone number; the phone number is part of the registered account for the device.

However a device which attaches to the network may not have a static IP assigned and will expect to get an IP address from a DHCP server on the network. When a cell phone or similar device equipped with a softphone application is attached to the network it is generally configured to receive an IP address from a DHCP server.

Where a MP network is attached to a LAN, there will usually be some device on the LAN running a DHCP server that will hand out a suitable IP address to an attaching device. As long as the MP static addresses are on the same sub net range as the DHCP addresses, all will be well.

Where a MP network is operating in a stand alone manner, not attached to a LAN, there will be no device present to hand out IP addresses. For this reason, a DHCP Server is provided in the firmware so that an MP device can perform this function.

The DHCP Server may be configured from the DHCP Server section of the Advanced SECN Configuration web page.

Care should be taken to avoid IP address conflicts, and conflicts between multiple DHCP servers on the same network. The range of addresses used for the DHCP server should be outside the range used for statically assigned addresses used for the MP devices.

### **Setting up the Softphone Clients**

For a Sipdroid client, the setup is as follows:

- Start up Sipdroid and go to the Sipdroid settings.
- Create a SIP account with Authorization Username set to one of the account entries in the file *softphone.sip.conf* (e.g. softph300),
- Set the Password to match the account entry e.g. "Pa55uu0rd300"
- Set the Server (or Proxy) to the IP address of the Softphone Master MP (ie .252 on the sub net)

Sipdroid should show successful registration to the softphone server.

### **Making Calls to and from Softphones**

To make a call from a softphone equipped device to an MP device simply dial the last octet of the MP's IP number in the usual manner.

To make a call from an MP device to a softphone device, simply dial the three digit number corresponding to the account entry e.g. "300"

Calls to softphone clients are only supported from MP devices with Softphone support set to "Client", and from the softphone Master device.



## 5.6 USB Extended File System

The SECN-1.1 firmware supports additional USB flash memory storage on devices that are equipped with USB ports. Examples of these devices include the TP-Link WR703N, MR3020, MR11U and WR842ND devices for which SECN-1.1 firmware has been ported.

USB drives on these are automounted to `/mnt` as normal **unless** they are labelled with one of two special volume names: **SECN-Extended** and **WEBSITES**. These labels enable two special purpose USB configurations which are used to support additional installed program packages and local web server content for the SECN-1.1 firmware.

### Extended filesystem for additional packages

The first pre-defined USB configuration requires the USB drive to be formatted as **ext3** and have a volume label of "**SECN-Extended**". Formatted and labeled this way, the USB drive will be automounted to `/user` instead of `/mnt`

There is a file "**SECN-extended.tgz**" available with the firmware which contains an extended filesystem for the device, including a pre-installed copy of Asterisk configured for use with SECN to support telephony in the same way as the MP-01 devices, including softphone support, but without the built-in ATA.

To set up a USB memory for this configuration, format the USB as **ext3** e.g.

```
$ mkfs.ext3 /dev/sda1
```

then label the drive "SECN-Extended" so that it gets automounted under `/user` e.g.

```
$ e2label /dev/sda1 SECN-Extended
```

Note: PLEASE be sure that you run **mkfs** only on your intended USB drive.  
A good way to check is to run:

```
$ cat /proc/partitions
```

and verify the drive's device node.

Alternatively you may use an application such as **Gparted** to format and label the USB device.

After formatting and labelling the USB flash drive, unpack the "**SECN-extended.tgz**" file into the root of the drive. The extended filesystem drive is now ready for use on the TP-Link SECN 1.1 device. With the TP-Link device turned off, insert the USB flash drive, and power up. If you have followed the steps correctly, the USB drive will be automounted to `/user`. Simply enter the **mount** command to verify.

**Note:** As of this writing, due to a bug in the OpenWRT automount feature, inserting the above drive in the TP-Link device **while it is running** will result in it being mounted to `/mnt` instead of `/user`. This won't hurt anything, but until you reboot, the features available on the extended filesystem won't be available due to the incorrect mount point.

### **Installing additional packages**

Other packages may be installed into this flash memory space with the command:

```
# opkg install -d usb <package-name>
```

### **Installing web content**

There is a directory called */websites* on this USB ext3 file system which may be used to store web content.

This directory appears as */user/websites* and is symlinked to */www/websites* on the device, so the content may be accessed through the web server at:

```
http://<ip-address>/websites
```

### **Using a VFAT USB for web content from Windows**

The second pre-defined USB configuration is formatted as the normal FAT32 (*vfat*) file system and has a volume label of "**WEBSITES**". This was done to more easily allow Windows users to capture websites to a USB drive since Windows support of the **ext3** formatted drive is limited.

This volume is mapped to */www/websites2* and so the web content will appear at:

```
http://<ip-address>/websites2
```

If you want to simply capture websites on a FAT32 USB drive (*vfat*), give it a volume label of "**WEBSITES**" and it will automount at boot up to */www/websites2*.

To capture websites on a USB drive under either Linux or Windows, a good free utility is **HTTrack Website Copier**.

There are both Linux and Windows versions at: <http://www.httrack.com>

**END OF DOCUMENT**



# Appendix

## **SECN-2.0 User Guide**

**Village Telco**  
**Small Enterprise / Campus Network**  
**SECN-2.0**

**User Guide**





SECN User Guide by T L Gillett is licensed under a  
[Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/).  
Based on a work at [www.villagetelco.org](http://www.villagetelco.org).

### **Acknowledgements**

This work would not have been possible without the contributions of many people associated with Village Telco.

In particular I would like to acknowledge the considerable contributions made by Elektra both in providing technical guidance and in building the software, as well as writing the text for sections of this manual.

Much input has also been provided by Keith Williamson, particularly in relation to development of the Softphone Support and DHCP features, and in building many test versions of the firmware.

I would also like to acknowledge the ongoing support and encouragement provided by Steve Song as a founder of the Village Telco project.

**Note:** This draft document is intended to be read in conjunction with SECN-2.0 firmware.

# Table of Contents

<b>1. Introduction.....</b>	<b>1</b>
<b>2. A Simple Mesh Set Up.....</b>	<b>2</b>
<b>3. Example Networks.....</b>	<b>4</b>
<b>4. Setting Up MP Devices.....</b>	<b>6</b>
4.1 Installing the SECN Firmware.....	6
Installing MP Firmware with the Potato-Flash Utility	
Installing with the sysupgrade Utility	
4.2 Minimum Set-up.....	10
Set the br-lan IP Address	
4.3 Set-up Using SECN Web Interface.....	12
Basic SECN Configuration	
Advanced SECN Configuration	
WAN Configuration	
Firmware Upgrade - MP-1 and AR23	
Firmware Upgrade - TP and AR71 Devices	
4.4 Advanced Set-up.....	27
Connecting to the device	
Setting the device Network Addresses	
Modifying Asterisk Operation	
Dial Plan for SIP / VoIP	
<b>5. Overview of SECN-1 Operation .....</b>	<b>30</b>
5.1 IP Address Range for MPs.....	30
5.2 Batman-Adv Operation.....	31
BATCTL Command	
bat-hosts file	
Batman-adv and Gateways	
5.3 Telephony Operation.....	34
Overview	
Interactive Voice Response (IVR) Commands	
IVR Command Summary	
5.4 Asterisk Operation.....	35
Making Calls to MP Devices	
Debugging Asterisk Operation	
Asterisk and Network Settings	
5.5 Softphone Support.....	39
Setting up the Devices	
Configuration of Softphone Accounts	
Setting up the DHCP Server	
Setting up the Softphone Clients	
Making Calls to and from Softphones	
5.6 USB Extended File System .....	42
Extended filesystem for additional packages	
Installing web content	
Using a VFAT USB for web content from Windows	



## **1. Introduction**

The VillageTelco Small Campus Enterprise Network (VT SECN) firmware is designed to allow a collection of Mesh Potato (MP) and similar devices (eg various TP-Link devices) with firmware based on OpenWrt, to provide a data and telephony network for a small campus or enterprise.

The intended use is typically for a small/medium size organisation which needs to set up a number of workpoints spread over a limited geographic area, with workpoints being equipped with a telephone and a networked PC, and to do this wirelessly without using conventional LAN cabling.

On a slightly larger scale, the system may also be used to provide networking for a small community, with shared access to network resources such as web server, file server and Internet access.

The meshed devices utilise an OSI Layer 2 protocol (batman-adv) and collectively simply act as one large switch, transparently connecting all the attached devices together.

Each MP device provides a telephone connection, an Ethernet cable connection, and a WiFi Access Point. TP devices provide mesh nodes without the telephone connection. PCs and other network devices may be connected to the Ethernet port of a mesh device, or connect wirelessly to the WiFi Access Point of each node.

The WiFi Access Point in each mesh node is encrypted with WPA by default in order to provide some protection from abuse of the data network as long as the pass phrase/key is kept confidential.

The Access Points may be configured to use the same SSID and password, in which case the WiFi 'cell' will effectively cover the same area as the mesh, and WiFi client devices will 'roam' throughout the cell. Alternatively, the Access Points may be individually configured so as to provide discrete WiFi cells.

If one or more of the mesh nodes is connected via its Ethernet port to a LAN with a router / DHCP server and Internet access, any device connected either by Ethernet cable to an MP or by WiFi, will be able to acquire a DHCP address on the LAN and connect to the Internet via the router.

Similarly, networked devices such as printers or storage devices may be attached to the LAN via a mesh node device. All attached devices will appear on the LAN and will be visible to each other.

Each MP device provides a telephone port which may be called from another MP telephone by dialling the IP address of the required device. Abbreviated dialling is also supported so that a call may be made by dialling just the last octet of the required IP address.

Support is also provided for Softphone applications running on smartphones, PCs or other devices.

To use telephony off the local mesh, individual mesh node devices can be configured to access a SIP/VoIP Service Provider account for outgoing and incoming calls.

Configuration and management of individual mesh node devices is possible via telephone IVR commands (MP only), browser or terminal sessions with access to the underlying OpenWRT Linux operating system and software.

## 2. A Simple Mesh Set Up

In this simple mesh network we will set up a network of two MP devices so that phone calls can be made between them, then connect one MP to a Local Area Network with Internet access so that a laptop can connect wirelessly to the virtual Access Point and access the LAN and Internet.

### Step 1

Flash the MP devices to the SECN firmware.

(See following section for details of how to flash the devices.)

### Step 2.

Set the unique IP address for each MP device.

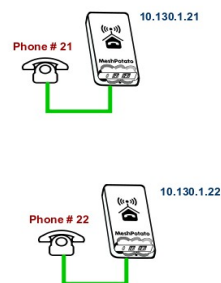
When the MP devices are rebooted, connect a telephone.

Lift the receiver and check for dial tone.

Dial **2662**, enter the pin **1234**, and when the announcement has finished dial **21**. Wait to hear the number being read back, then reboot the device when prompted.

Repeat the process with the second MP, but dial **22** and wait for it to reboot.

The MP devices are now set to IP addresses **10.130.1.21** and **10.130.1.22** respectively. It may be useful to label the devices as '21' and '22'



### Step 3

Make a phone call.

After the MP devices have fully rebooted (allow a couple of minutes after the WiFi light starts to flash), pick up the phone on the '21' MP, check for dial tone and dial **22**. The other phone should start to ring after a few seconds. Repeat the other way around.

### Step 4

Attach the mesh network to your LAN.

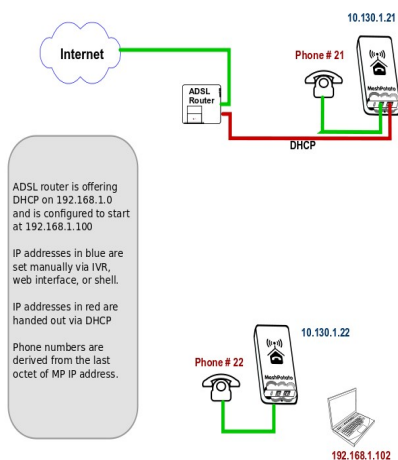
Connect the MP '21' to a spare port on your router with an Ethernet cable. The diagram shows the LAN using an IP address range of **192.168.1.xxx**, but the actual range used will not matter

*Note:— Because the mesh utilises an OSI Layer 2 protocol, it will work with any LAN address range. The MP addresses do **\*not\*** have to be in the same subnet as the LAN in order for the mesh to carry LAN data traffic.*

### Step 5

Attach a laptop via WiFi.

Your laptop should be able to see a WiFi Access Point called **VT-SECN-AP** secured with WPA encryption. Connect to this Access Point with a WPA password of '**potato-potato**' and using Automatic assignment of IP address (DHCP).



Your laptop should acquire an IP address in the range offered by your LAN router, and you should be able to access the Internet.

You should be able to make calls between the MP devices while accessing the Internet on the laptop.

You can connect another PC to the '22' MP using an Ethernet cable and it will similarly acquire an IP address from the router.

The laptop and PC should be able to access any other devices on the LAN, such as printers or network storage devices just as if they were connected directly to the LAN.

### 3. Example Networks

Following are examples of practical networks built around MP devices operating in a mesh.

#### Network 1

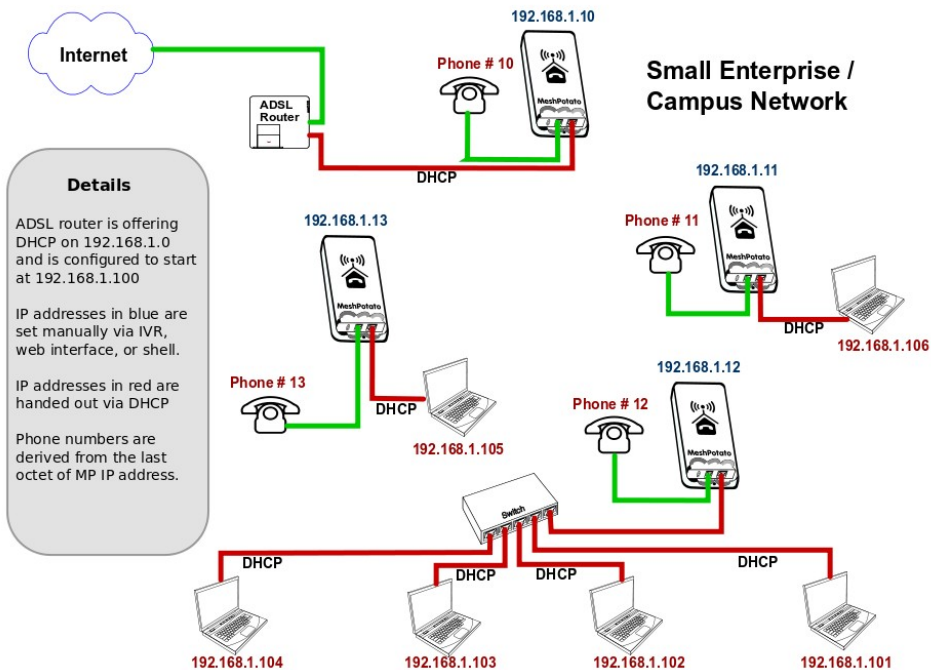
In this network, MP devices have been assigned static IP addresses that are part of the LAN address space, **192.168.1.xxx**, and appropriate Gateway and DNS addresses.

This means that the MP administration interfaces (SECN web interface and **ssh** command line) will be accessible from any workstation connected to the LAN.

When a workstation is attached to the mesh network either by Ethernet cable or WiFi, it will acquire an IP address from the router in exactly the same manner as if it was connected directly to the router.

The router address space must be managed so that there is no conflict between the statically assigned MP addresses and those for any other device on the network. In this example the router offers DHCP addresses starting at **192.168.1.100**, while the MPs have been assigned static addresses below this range.

Phone calls may be made between MP devices either by dialling the last octet of the required MP address (e.g. '10'), or by dialling the full address (e.g. '192\*168\*1\*10').



## Network 2

In this network, MP devices have been assigned static IP addresses that are not part of the LAN address space. Instead they have been assigned IP addresses in the default address space 10.130.1.xxx.

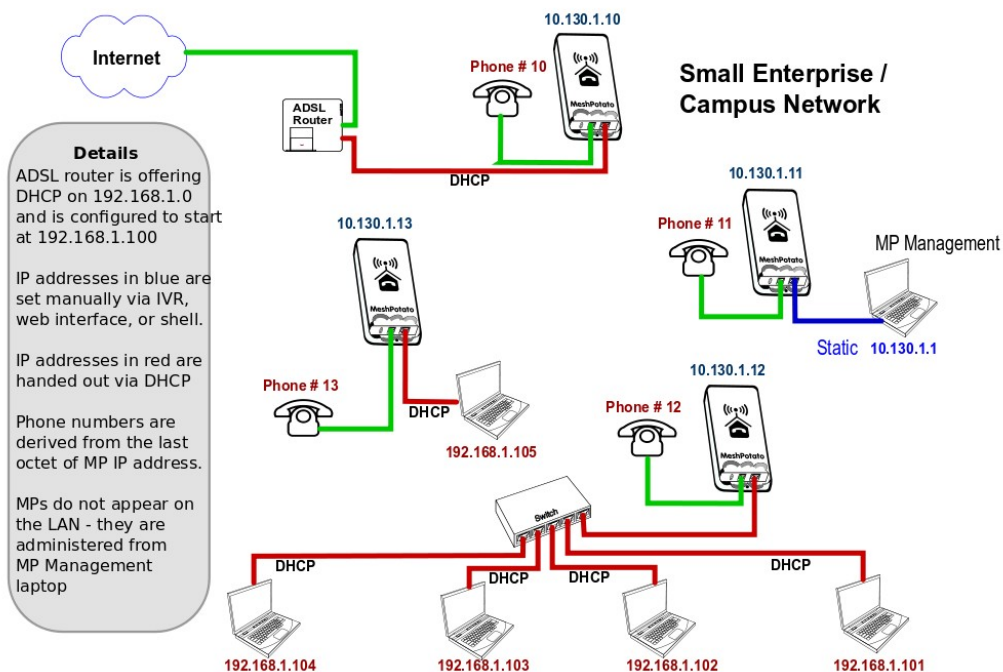
This means that the MP administration interfaces (SECN web interface and *ssh* command line) will not be accessible from workstations connected to the LAN with IP addresses assigned in the LAN address space.

Administration of the MP devices may be undertaken from a workstation assigned a static address in the same range as the MP devices and attached via Ethernet cable or WiFi to any MP device in the network.

When a workstation is attached to the network either by Ethernet cable or WiFi, it will acquire an IP address from the router in exactly the same manner as if it was connected directly to the router.

In this example there is no need to manage the LAN address space to allow for the MP addresses as they are allocated in a completely different address space (10.130.1.xxx).

Phone calls may be made between MP devices either by dialling the last octet of the required MP address (e.g. '10'), or by dialling the full address (e.g. '10\*130\*1\*10').



## 4. Setting Up MP Devices

This section describes how to set up MP or TP devices for use on your mesh network.

The first step is to install the SECN firmware on the MP or TP device.

After installing the firmware, three different methods are available to configure the device:

- **Minimum Setup** using telephone Interactive Voice Response (IVR – MP only)
- **Basic Setup** using the SECN web browser interface.
- **Advanced Command Line Setup** using a *ssh* terminal session and command line.

### 4.1 Installing the SECN Firmware

If you have purchased a new MP-1 device, it may be delivered from the factory with VT firmware version rv233 installed. To operate with the SECN configuration, you will need to flash the MP with the appropriate SECN firmware. Similarly you may wish to upgrade the SECN firmware version.

There are two methods for installing the firmware:

- Use the *Potato-Flash* utility on a Linux PC connected to the MP via Ethernet cable.
- Use the *sysupgrade* utility from the command line on the MP/TP device.

Using the *sysupgrade* utility has the advantage that the firmware can be installed on the MP/TP 'over the air' without the need to connect with an Ethernet cable, which may avoid the need to physically recover the device from an installation.

**NOTE:** *Early MP devices may not be immediately suitable for flashing with the sysupgrade utility until they have been flashed at least once with the potato-flash utility. This is due to an incorrect memory layout from a different flashing program. If possible, use the potato-flash utility as the preferred way to flash the MP device.*

To check the status of your MP, run the command:

```
cat /proc/mtd
```

The correct output looks like:

```
dev:      size      erasesize  name
mtd0: 00030000 00010000 "RedBoot"
mtd1: 000b0000 00010000 "vmlinux.bin.17" Note that mtd1 contains the Linux kernel
mtd2: 006f0000 00010000 "rootfs"
...
```

An incorrect output may look like this:

```
dev:      size      erasesize  name
mtd0: 00030000 00010000 "RedBoot"
mtd1: 006f0000 00010000 "rootfs" Note that mtd1 does not contain the Linux
kernel
mtd2: 00410000 00010000 "rootfs_data"
...
```

### Installing MP Firmware with the Potato-Flash Utility

These instructions assume that you are running Ubuntu or other Linux distribution on your PC. If this is not the case, use one of the other methods to flash the device.

#### 1. Set up the *potato-flash* application on your PC

Download the potato-flash file from:

<http://villagetelco.org/download/utilities/>

Save the file into */usr/local/bin*

Make the file executable:

```
chmod +x /usr/local/bin/potato-flash
```

#### 2. Download the firmware

Download the required firmware from:

<http://villagetelco.org/download/firmware/secn/>

Download the *.squashfs* and *.lzma* files for the required firmware version and save to a working directory.

#### 3. Set up networking on your PC

This step will ensure that *potato-flash* has proper access to the PC Ethernet network port.

Connect the MP directly to your PC with an Ethernet cable **with the MP power off**.

In Ubuntu Gnome desktop, right click on the Network Manager icon and deselect **Enable Wireless**

Left click on the Network Manager icon and **Disconnect** any **Wired Network** that is active.

#### 4. Flash the MP

Following is a brief description of the flashing process. Refer to the general instructions in *Upgrading Mesh Potato Firmware HowTo* on the Village Telco Wiki for more detail.

- Connect the MP directly to your PC with an Ethernet cable with the MP power **off**
- Execute potato-flash:

```
$ sudo potato-flash eth0 <filename>.squashfs <filename>.lzma
```

- Assuming the Ethernet port on the PC is *eth0*.
- Note that the order of the *.squashfs* and *.lzma* files is mandatory in the command.
- Enter your password when prompted.
- Wait for the program to start looking for the MP device - a series of dots will appear on the screen.
- Switch the power **on** to the MP.
- Wait for the flashing process to complete and for the MP to fully restart.  
**This may take a couple of minutes.** This is a good time to have a coffee.
- Wait for three minutes after the MP WiFi led starts to flash to ensure that flashing process is complete. Some early MP devices may take quite a long time (10mins +) to load and flash.

### Sample MP Potato Flash Session

```
$ sudo potato-flash eth0 openwrt-atheros-root-rv238.squashfs openwrt-atheros-vmlinux-rv238.lzma
Reading rootfs file openwrt-atheros-root-rv238.squashfs with 3801088 bytes ...
Reading kernel file openwrt-atheros-vmlinux-rv238.lzma with 720896 bytes ...
Note: The device has to be connected directly (not via switch/hub)
Device detection in progress.....
```

<<< *Turn the power to the MP device ON at this point* >>>

```
.....device detection: non-arp packet received..
Peer MAC: 00:09:45:58:1c:e7
Peer IP : 192.168.1.184
Your MAC: 00:ba:be:ca:ff:ee
Your IP : 192.168.1.0
Connecting to Redboot bootloader
WARNING: UNPLUGGING POWER WHILE FLASHING MIGHT DAMAGE THE BOOTLOADER
HOWEVER: IF YOU SEE NOTHING SHOWING UP BENEATH THIS LINE
FOR MORE THAN A MINUTE, START AGAIN...
A flash size of 8 MB was detected.
rootfs(0x006a0000) + kernel(0x00100000) + nvram(0x00000000) sums up to 0x007a0000 bytes
Setting IP address...
Initializing partitions...
Now uploading kernel...
Sending kernel, 1408 blocks...
Flashing kernel...
Loading rootfs...
Sending rootfs, 7424 blocks...
Flashing rootfs...
Flashing process completed...
Restarting device...
```

### Installing with the *sysupgrade* Utility

To install with the *sysupgrade* utility on the MP or TP device, it is necessary to copy the required *.img* file to the MP/TP using the *scp* command from within a *ssh* session on your PC. You may also use *sftp* to browse the unit's file system in Nautilus or with WinSCP.

An MP/TP device flashed with SECN firmware will only provide terminal access via *ssh* by default using the login account of *root* once the system password has been set..

If you are flashing a new TP device running the original factory firmware, you will need to use the '*factory*' version of the firmware *.img* file, rather than the one *sysupgrade* version of the *.img* file.

This is required only for the first time the device is flashed to the VT SECN firmware. Use the IP address and web interface of the manufacturer's firmware to load the new firmware file.

If you are re-flashing a TP device that already has the VT SECN or other OpenWrt firmware version loaded, follow the process outlined below using the *sysupgrade* version of the firmware.

If you are using a new MP it will operate with IP addresses of [10.130.1.20](#) (LAN) and [172.31.255.254](#) (Fallback). To use one of these addresses, configure your PC Ethernet networking profile with a static address to be able to access either of these addresses, and connect directly with an Ethernet cable to the MP device.

For example, to use the MP Fallback IP address, set the PC network profile to:



IP: **172.31.255.253** Netmask: **255.255.255.252** (Note restricted IP and Netmask values)

Alternatively you may set the MP device address to work on your LAN. Connect a telephone to the MP and dial the IVR command **C-O-N-F** (2663) and follow the prompts to set the IP number to one that lies in your normal LAN address range and is not already in use. The device will then reboot. Connect the MP device to an Ethernet port on your LAN and it will be accessible by any PC on the LAN.

From a terminal session on your PC, transfer the required **.img** file to the MP using the **scp** command e.g

```
scp ./openwrt-atheros-root-rv287.img root@172.31.255.254:/tmp
```

This will place the file in the **/tmp** directory on the MP device. Note that the contents of **/tmp** are stored in volatile RAM and thus will be lost on a system restart.

From the **ssh** session install the firmware with the command:

```
sysupgrade -n -v ./<filename>.img
```

The flashing process will begin and may take several minutes, after which the MP device will restart.

Note that the **-n** flag causes previous configuration settings **not** to be retained i.e. the device will operate with the default setting after the flash. This may be an issue for remotely accessed devices – see later section for discussion on this.

After the MP device has restarted and the WiFi led has started to flash, allow up to three minutes for the flashing process to complete. After that you should be able to connect to the MP device with web browser or **ssh** on the default LAN or Fallback IP addresses.

You may also use the IVR **C-O-N-F** (2663) command to change the MP device address to work on your LAN.

## 4.2 Minimum Set-up

The Minimum Set-up process uses the telephone IVR facility to simply set a unique IP address for the br-lan bridge interface in order to allow telephone calls to the device using the IP address. Even with this minimal configuration, the MP mesh network may be connected to a LAN and will provide WiFi and Ethernet connectivity for PCs and other devices to the LAN and Internet.

The default setting for the br-lan IP address when the device is flashed is **10.130.1.20** and you should change at least the last octet of the address in order to make the address unique on the mesh to support telephone dialling.

In a simple mesh arrangement, all MP devices on the mesh are assigned addresses in the same address range (ie only the last octet of the address is changed) so telephone calls can be made to all devices on the mesh with abbreviated dialling using just the last octet of the MP device's bridge IP address.

If you are intending to connect the mesh to a LAN, you may choose to assign addresses from the LAN address space to the MP devices so that they will appear as static IP devices on the LAN.

In this case, just set the IP address of the MP device to the required IP address on the LAN.

You will need to ensure that the address that has been assigned will not be used by any other device on the LAN in order to avoid IP conflicts.

### Set the *br-lan* IP Address

Connect a telephone to the MP device and check that you have dial tone.

Use one of the methods below to set the device address.

#### Set Abbreviated Address:

- Pick up the telephone, check for dial tone and dial 2662
- Enter the IVR Pin number (default 1234)
- Follow the voice prompts and enter the required number for the device as 1 – 3 digits
- e.g 21. This will set the last octet of the MP device IP address e.g. 10.130.1.21
- The number entered will be read back to you and a prompt to reboot the MP.
- The command will fail if the IP is in use (ping) or out of range (.1 to .254).

#### Set Full IP address:

- Pick up the telephone, check for dial tone and dial 2663 (C-O-N-F)
- Follow the voice prompts and enter the IP number in the form 10\*130\*1\*21
- (For an IP address of 10.130.1.21)
- The number entered will be read back to you and a prompt to reboot the MP.
- The command will fail if the IP is in use (ping).

After the device has rebooted, you should be able to make a call to the device using either the full IP number, or abbreviated dialling using just the last octet of the address.


**Note:** When the *br-lan* address is set using IVR, the device's *gateway* address will be automatically be set to an address in the same subnet with the last octet set to **1** e.g 10.130.1.1 to ensure correct operation of Asterisk.

If you plan to connect the mesh devices to a LAN and you use this method to set up the MP to have

an address in the LAN address space, then the MP will expect to find your LAN router at the *x.y.z.1* address. If your router has a different address, you may use the 4283 (G-A-T-E) IVR command to change the *gateway* address as required after setting the IP address.

## 4.3 Set-up Using SECN Web Interface

### Basic SECN Configuration



**SECN Configuration**  
Firmware: Version 2.0 MeshPotato-1  
Date: Wed May 1 10:00:47 AEST 2013

Basic
Advanced
Status

**Network**

IP Address

Gateway

Find Gateway

**WiFi Access Point (WPA1)**

Station ID

Passphrase

Channel

**VoIP / SIP Configuration**

User Name   
SIP Host   
SIP Enable ☐

Password   
Dialout Code   
SIP Status Not Registered

**Password**

Enter Password

Repeat Password

**Web Server Security and Timezone**

Limit IP Address ☐

Enable SSL ☐

Time Zone

The Basic SECN Configuration screen may be accessed by pointing your web browser to the IP address of the MP device. A newly flashed device will not have a root password set and thus the web interface will not require authentication.

For a newly flashed device you may use the default IP address of 10.130.1.20 or the Fallback IP address of 172.31.255.253 To use either of these addresses you will need to configure networking on your PC to be able to access these subnets.

Alternatively you may wish to first set the IP address of the MP so that it appears on your LAN subnet by using the phone IVR menu as described in the previous section. If doing so, make sure that you assign an IP address that does not conflict with other devices on the network.

The Basic SECN Configuration screen allows you to set up just the key parameters for Network Address, Gateway, WiFi Access point, a SIP/VoIP phone service, set the password for the root account, and configure the web server security.

A link is provided at the top of this screen to allow access to the Advanced SECN configuration screen if required.

### Network Configuration

The network configuration parameters that can be set up are the IP Address for the MP device and the IP address for the Gateway (router) device on the local network which provides access to the Internet.

The Find Gateway button will attempt to locate the Gateway device by sending a DHCP Discover request on the network. If a device responds to the request, then the address of the responding device will be shown in a status message at the bottom on the page. Enter the required Gateway device address in the field and click on the Save button.

### WiFi Access Point Configuration

The WiFi configuration allows you to set the Station ID (SSID), Passphrase and radio Channel for the MP device.

The Station ID must be comprised of alphanumeric characters (plus dash and underscore). This is the name of the WiFi Access-point that will be seen from a WiFi client device attempting to connect.

The Passphrase will be required to allow a client to connect if WiFi encryption is being used. The Passphrase must be comprised of alphanumeric characters (plus dash and underscore) and must be a minimum of eight characters in length.

Note that as this is the only security that prevents wireless access to your network, you should change the default password and make it suitably strong by using a combination of upper and lower case letters, and numbers.

By default the SECN firmware operates using WPA1-PSK encryption on the WiFi access point. You may change the encryption if required on the Advanced screen.

### VoIP Configuration

The VoIP configuration allows you to set up a SIP/VoIP service to make calls beyond the local mesh if you wish. Typically this is via a commercial VoIP service provider that provides access to the standard telephone network.

Note: For this to work, the MP device must be correctly configured to operate on a network which provides Internet access for the device through the specified Gateway. In particular, the MP must be configured with an IP address in the same subnet as the Gateway IP address.

The settings are used to update the `/etc/asterisk/sip.conf` file via the `potato.sip.conf` file.

When you establish an account with a SIP/VoIP provider, you will be given a User Name and Password, as well as the URL of the SIP server on the Internet. Enter these details in the relevant fields on the screen. The Password will only be displayed when first entered.

The Dialout Code is used to distinguish between calls to be made on the local mesh, and those to be routed out to the SIP/VoIP provider. The specified digit needs to be dialled before the required external number. The available digits are #, 0 and 9.

The SIP Enable checkbox allows you to enable or disable the external SIP service while keeping the saved settings.

After entering the required settings, click the Save and Restart Asterisk button. If the MP can successfully contact the SIP server and register, the registration status will be shown below the Dialout control. Note that registration may take some time and the status may not show

immediately. You can click the Refresh button to check the status after a period of time.

### Password

These fields allow the password to be changed for the root account by default.

After entering the password in both fields, click on the Set Password button to make the change.

A status line at the bottom of the page will indicate whether the change was successful.

### Web Server

These controls provide access security configurations to be applied to the web based configuration screens. The options can be applied in any combination and require a restart to become effective.

The Limit IP Address checkbox restricts access only to the Fallback IP address [172.31.255.254](#) with Netmask [255.255.255.252](#)

A connecting PC will need to be set to an IP address of [172.31.255.253](#) in order to gain access.

The Enable SSL checkbox makes access to the unit only available using SSL, thus encrypting data over the link.

When used for the first time, the unit generates a self-signed certificate, which the web browser on a connecting PC will flag and require the user to accept the certificate before allowing access.

When SSL is enabled, the required URL is: `https://<ip-address>`

### Saving and Rebooting


The Refresh button will simply refresh the field values from saved values in memory and check the registration status of the SIP/VoIP service.

The Save button writes the various field values on the screen into memory, but does not change the operation of the MP device.

The Save and Restart Asterisk button will save all field values and restart the Asterisk telephony subsystem. This usually takes about 10 seconds to complete, but may take longer if Internet access is not configured correctly.

The Save and Reboot button will save the field values and restart the MP device using the newly saved values. Note that a restart is required to effect changes to the Network, WiFi and Web Server security settings, and will take around two minutes to complete.

Advanced SECN Configuration



SECN Configuration

Firmware: Version 2.0 MeshPotato-1

Date: Wed May 1 10:00:47 AEST 2013

BasicAdvancedStatus

AdvancedWANFirmware

Time: 10:00:47 up 15:52, load average: 3.40, 1.18, 0.58 TZ: AEST-10

Network

IP Address10.130.1.20Gateway10.130.1.1

DNS8.8.8.8Netmask255.255.255.0

Radio

Channel1US/Can (11 ch)☐Tx Power 1-2017

Wifi Mode802.11GChan BW20

WiFi Access Point

SSIDVT-SECN-AP

Passphrasepotato-potatoEncryptionWPA1AP Connections20

WiFi Mesh

IP Address10.10.1.20Netmask255.255.255.0

SSIDvt-meshBSSID02:CA:FF:EE:BA:BE

Gateway ModeOFFEncryptionOFF

Asterisk Configuration

The Advanced SECN Configuration screen may be accessed by clicking on the link at the top of the Basic SECN Configuration page.

This screen allows you to set up basic and additional parameters for Network, WiFi, a SIP/VoIP phone service, Softphone support and DHCP server.

Links are provided at the top of this screen to allow access to the Basic SECN and Wireless Status configuration screens if required.

Network Configuration

The network configuration parameters that can be set up are the IP Address and Netmask for the MP device, the IP address for the Gateway (router) device on the local network, which provides access to the Internet, and the IP address of the DNS server to be used for name resolution.

Radio Configuration

The XXXXXXXXXXXXXXXXXXXXXXXXXX

WiFi Access Point Configuration

The WiFi configuration allows you to set the SSID (Station ID), Passphrase, Encryption and radio Channel, and the maximum number of connections for the MP device.

SECN-2.0\_UserGuide-d1

15

The **US/Can (11ch)** checkbox sets the regulatory domain for North America to limit the number of available channels to 11 in accordance with FCC regulations. When this mode is active and channel 12 or 13 is selected, the channel setting will be set to Channel 1.

The **SSID** must be comprised of alphanumeric characters, plus dash and underscore. This is the name of the WiFi Access-point that will be seen from a client device attempting to connect.

The **Passphrase** will be required to allow a client to connect if WiFi encryption is being used. The passphrase must be comprised of alphanumeric characters (plus dash and underscore) and must be a minimum of eight characters in length. Note that as this is the only security that controls access to your network, you should change the default password and make it suitably strong by using a combination of upper and lower case letters, and numbers.

The **Encryption** control allows you to select from WPA1, WPA2, WEP or Open encryption on the WiFi Access-point.

The **AP Connections** control sets the maximum number of WiFi associations that will be supported. This may be used to manage the load on APs in an extended WiFi cell arrangement, or to disable the AP.

### WiFi Mesh Configuration

The Mesh Wireless configuration allows you to set a number of parameters for the mesh `ath0` interface including: **IP Address**, **Netmask**, **SSID**, **BSSID**, **Transmit Power**, and Country Code. These values are written into the configuration files `/etc/config/network` and `/etc/config/wireless`.

The `ath0` interface used for the mesh wireless protocol has an IP Address and Netmask. These are set to default values of `10.10.1.20` and `255.255.255.0` and normally do not need to be altered. These settings are not used for the OSI Layer 2 Batman-adv mesh protocol, and so all MP devices on the mesh may remain on the default IP address.

The `ath0` IP address can be used to access the MP device for maintenance in the same way as the Network Address or the Fallback Address described previously. If it is intended to use this address for maintenance access, it should be set to a unique value to avoid any potential IP address conflict.

The **SSID** and **BSSID** parameters set the station identification for the MP on the mesh and should be set the same for all devices in a mesh cell. These parameters can be used to set up separate mesh cells if required.

Note: It is a requirement of the current OpenWrt operating system that the **BSSID** must commence with an even number eg 02, 04, 06 etc.

The **Tx Power** parameter may be used to adjust the power of the device radio transmitter. It is set by default to the maximum value of 17. Normally this should not need to be adjusted but doing so may be useful in certain circumstances such as testing.

The **Encryption** control may be used to enable encryption on the mesh, if the device supports it.

The **MP Gateway Mode** determines whether the device will act as a Gateway in the Batman-adv mesh routing protocol. This setting is only needed if there is more than one gateway device on the mesh. A device which is connected to a LAN in order to provide Internet access for example, should be set to Server mode to assist routing requests efficiently through the mesh. Devices requiring access through a Gateway should have this set to Client mode.

The **WiFi Mode** control allows selection of the hardware modes supported by the device



eg 802.11G and 802.11N-G.

**WIFI Mesh**

IP Address: 10.10.1.20      Netmask: 255.255.255.0

SSID: vt-mesh      BSSID: 02:CA:FF:EE:BA:BE

Gateway Mode: OFF      Encryption: OFF

**Asterisk Configuration**

Enable Asterisk: ☒      Softphone Support: OFF

Codec1: gsm      Codec2: ulaw      Codec3: alaw

SIP Enable: ☐      SIP Register: ☐      Dialout Code: #

SIP Status: **Not Registered**

SIP Registrar: sip.myhost.com      User Name: myuser

SIP Host: sip.myhost.com      Password: \*\*\*\*

Enable Asterisk NAT: ☐      NAT External IP: 0.0.0.0

**DHCP Server**

Enable DHCP Server: ☐      Authoritative: ☐

Starting IP: 10.130.1.200      Ending IP: 10.130.1.240

Subnet Mask: 255.255.255.0      Gateway Router: 192.168.1.1

Lease Term (secs): 7200      Max Leases: 40

Domain: lan

Refresh   Save   Restart Asterisk   Restore Defaults   Reboot

## Asterisk Configuration

The VoIP / SIP configuration allows you to set up a SIP/VoIP service to make calls beyond the local mesh if you wish, and to optionally support Softphone operation on devices attached to the mesh.

Typically VoIP / SIP operation is via a commercial VoIP service provider that provides access to the standard telephone network.

Note: For this to work, the MP device must be correctly configured to operate on a network which provides Internet access for the device through the specified Gateway. In particular, the MP must be configured with an IP address in the same subnet as the Gateway IP address.

The settings are used to update the `/etc/asterisk/sip.conf` file via the included `potato.sip.conf` file.

When you establish an account with a SIP/VoIP provider, you will be given a **User Name** and **Password**, as well as the URL of the **SIP Host** and **Registrar** server on the Internet. Enter these details in the relevant fields on the screen.

The **Enable Asterisk Nat** checkbox may be used to enable Asterisk use behind a NAT firewall. Normally this is not required for a LAN behind a simple router/NAT firewall providing Internet access, but may be required, for example, if the MP is behind a second NAT firewall. If used, the External NAT IP field should be set to the upstream network IP address of the NAT router to which

the MP is connected.

The [Dialout Code](#) is used to distinguish between calls to be made on the local mesh, and those to be routed out to the SIP/VoIP provider. The specified digit is required to be dialed before the required external number. The available digits are #, 0 and 9.

The [SIP Enable](#) checkbox allows you to enable or disable the external SIP service while keeping the saved settings.

The [Register](#) checkbox determines whether the device will register with the SIP host. Registration is required in order to receive incoming calls, and some providers require registration for outgoing calls as well.

The Softphone Support control is used to set the mode of operation of the MP device in conjunction with other devices such as cell phones or laptops attached to the network typically via WiFi. See later section for details of Softphone operation.

NOTE: One device only on the mesh may be set to Master mode, and this device will automatically be configured to use the reserved IP address of .252 on the LAN segment in use.

The [Codec](#) settings may be used to control the Codecs available to be used for calls. Normally this will not need to be changed, however some SIP/VoIP providers do require specific codecs to be used, in particular for calls outside their immediate domain.

After entering the required settings, you may click the Save and Restart Asterisk button for the changes to be made effective. If the MP can successfully contact the SIP/VoIP server and register, the registration status will be shown alongside the Sip Status label.

Note that registration may take some time and the registered status may not show up when the screen is first refreshed. You can click the Refresh button to check the status.

### DHCP Server Configuration

The DHCP configuration allows you to set up a DHCP server to operate on the device. This may be used, for example, to ensure that devices attaching to the mesh network are able to obtain an IP address via DHCP in the event that there is no service available from a gateway device, perhaps due to the absence of an uplink to a remote router device.

Note: Care must be taken in setting up the configuration of the DHCP server to ensure that there is no conflict between multiple DHCP servers that are visible to devices attached to the network. Normally only a single DHCP server is enabled on a network.

The DHCP server provides IP address leases and a range of network information to clients in response to a DHCP Discovery request. The settings for the DHCP server that can be configured from the MP device web interface are outlined below.

The [Enable DHCP Server](#) checkbox allows the server in the MP device to operate when it is checked. By default the DHCP server is not enabled.

The [Starting and Ending IP](#) fields set the range of addresses that will be handed out by the DHCP server. Care must be taken to ensure that this range does not overlap the range of any other DHCP server on the network.

[Lease Term](#) sets the time period in seconds that IP address leases are valid.

[Max Leases](#) sets the maximum number of concurrent leases that will be handed out.

**DNS** defines the Domain Name Server IP addresses that will be handed out to clients as part of the DHCP protocol.

**Domain Name** sets the network name that will be handed out to clients as part of the DHCP protocol.

**Subnet Mask** sets the network mask that will be handed out to clients as part of the DHCP protocol.

**Router** sets the IP address of the network gateway that will be handed out to clients as part of the DHCP protocol.

## WAN Configuration

Date: Wed May 1 10:14:41 AEST 2013

Basic **Advanced** Status

Advanced **WAN** Firmware

**WAN Configuration**

WAN Port  Note: If a WiFi WAN port is selected, Mesh and AP interfaces are disabled on that port.

WAN IP Mode

**Static Network Settings**

Static IP  Gateway

Netmask  DNS

**WiFi WAN Host Settings**

SSID

Passphrase  Encryption

**USB Modem Settings**

USB Modem Service

Vendor ID  Product ID

Service APN  Dial String

Username  Password

PIN  USB Serial Port

USB Device Detected **USB2.0 Hub Vendor=05e3 ProdID=0608**

USB Serial Ports Detected

USB Modem Status

## WAN Port

WAN Configuration allows the router to be configured with one of its network ports acting as a WAN port, with Network Address Translation (NAT) in operation between the WAN port and the other network ports attached to the internal bridge. By default, the WAN interface is **Disabled**.

### *Ethernet WAN*

Selecting **Ethernet** for the WAN interface will make the Ethernet port act as a WAN port.

On devices (e.g. TP Link routers) with multiple Ethernet ports, the port designated as the WAN port (often coloured differently to the others) will be made the WAN port, with the others remaining connected to the internal bridge. In WAN Disabled mode, this port will be inactive.

### *WiFi WAN*

Selecting **WiFi** for the WAN interface will disable the WiFi Access Point and Mesh interfaces, and makes the router act as a WiFi Station that will attach to an Access Point as specified in the **WiFi WAN Host Settings** section.

**Note:** Because the WiFi mesh interface is disabled in this mode, the device will **not** be part of the mesh network. This is a limitation of the OpenWrt wireless drivers at this time.

### ***USB Modem WAN***

Selecting ***USB Modem*** for the WAN interface allows the use of common USB modems

#### **WAN IP Mode**

This setting determines whether the WAN interface will operate as a ***DHCP*** client, obtaining its IP address from the network to which it is connected, or whether it will have a ***Static*** IP address.

#### **Static Network Settings**

These settings are used to configure the WAN interface if ***Static*** mode is selected.

#### **WiFi WAN Settings**

These settings are used to specify the details of the Access Point to which the device will attach if WiFi WAN mode is selected, including the ***SSID*** and ***Encryption*** mode and ***Passphrase***.

#### **USB Modem Settings**

These settings are used to configure the USB Modem for devices that have a USB port.

##### ***USB Modem Service***

Select the value corresponding to your wireless broadband service, either UMTS or XXX

##### ***Vendor ID and Product ID***

These settings need to match the values for the modem hardware. If a modem is plugged in and the device restarted, the values will be displayed in the status line ***USB Device Detected***.

These are four character hexadecimal values.

Note that there may be multiple values shown for devices with more than one USB port, and you need to select the values corresponding to the modem, and enter them into the appropriate fields.

##### ***Service APN***

This is the APN value for the wireless broadband service and will be provided by the service provider.

##### ***Dial String***

This is the dial string for the wireless broadband service and will be provided by the service provider.

##### ***Username, Password***

These values may be required by your service provider. If not required, leave them blank.

##### ***PIN Number***

If the PIN Number has been activated for your modem, enter the value here.

##### ***USB Serial Port***

The USB Serial Port number is specific to the USB modem device.

For example, Huawei devices generally use 0, and Sierra Wireless generally use 2.

Once the ***Product*** and ***Vendor ID*** values are correctly set, and the device is restarted with the USB modem installed, the USB Serial ports detected will be displayed in the status line at the bottom of the page.

When the USB modem settings have been correctly entered and the device restarted with the modem installed and the Wireless Broadband service is available, the connection status will be displayed in the status line at the bottom of the page.

Firmware Upgrade - MP-1 and AR23

For the Mesh Potato 1 and Ubiquiti devices based on the AR23 chipset, this page allows you to upload the 'root' and 'vmlinux' firmware files in order to reflash the device.

After selecting *Firmware Upgrade* and *Proceed with upgrade*, browse for and select the new firmware files, then select *Upload files to server*.

The files will be uploaded and the flash process started. A screen will be displayed showing the time remaining for the upgrade to be completed. This is typically five minutes, and it is important not to disrupt the process during this time.

[Return to Configuration](#)  
**Upgrade your Mesh Potato firmware**  
If you click "Proceed with Upgrade" the upgrade process will be begin by closing down any non-essential running programs. This includes telnetd, sshd, hostapd, ntpd and others. This is done in order to free sufficient ram for the upgrade. If even you do not proceed after the next stage, you will still need to reboot in order to restore full functionality to the Mesh Potato.  

Proceed with upgrade

Shutting down telnetd...  
Shutting down sshd...  
Shutting down ntpd...  
Shutting down asterisk...  
Shutting down wireless AP...  
Shutting down misc...  
Free memory is...

	total	used	free	shared	buffers
Mem:	13240	12112	1128	0	1016
-/+ buffers:		11096	2144		
Swap:	0	0	0		

**Upgrade your firmware**  
Filename (vmlinux) 

Browse...

  
Filename (root) 

Browse...

Upload Files to Server

Progress  

0%

Please be patient. There may be a delay even after the progress bar reaches 100%



## Firmware Upgrade - TP and AR71 Devices

For TP Link, Ubiquiti and other devices based on AR71 and compatible chipsets, this page allows you to upload a 'sysupgrade' firmware file in order to reflash the device.

After selecting **Firmware Upgrade**, browse for and select the new firmware file.

It is preferable to also enter the MD5 checksum of the file to ensure that it has not been corrupted, but you may choose to skip this feature by checking the **Ignore Checksum** box.

Select **Upload File to Server** to transfer the file and checksum to the device.

The file will be uploaded and the MD5 checksum calculated and compared to that supplied.

[Return to Configuration](#)

**Upgrade your firmware**

Filename:

Ignore Checksum? ☐

Paste checksum:

Progress:  0%

If the checksum comparison is correct, you may select whether to preserve the current device configuration (eg IP address, SSIDs, passwords etc), then select **Upgrade Firmware** to begin the flash process.

A screen will be displayed showing the time remaining for the upgrade to be completed. This is typically five minutes, and it is important not to disrupt the process during this time.

[Return to Configuration](#)

**Upgrade your firmware**

Filename:

Ignore Checksum? ☐

Paste checksum:

Progress:  100%

You uploaded **openwrt-ar71xx-generic-tl-wr703n-v1-squashfs-sysupgrade.bin**.

The checksum of the uploaded file is: c8904bb9a201b99969734c2c74194900  
 The checksum you submitted is: c8904bb9a201b99969734c2c74194900  
 Congratulations your checksums match. The file uploaded correctly.

Preserve existing configuration?  
☐ Yes  
☒ No

### **Saving and Rebooting**

The [Refresh](#) button will simply refresh the field values from saved values in memory and check the registration status of the SIP/VoIP service.

The [Save](#) button writes the various field values on the screen into memory, but does not change the operation of the MP device.

The [Save and Restart Asterisk](#) button will save all field values and restart the Asterisk telephony subsystem. This usually takes about 10 seconds to complete, but may take longer if Internet access is not configured correctly.

The [Restore Defaults](#) button will reset all the configuration settings to the default values of a newly flashed device.

The [Save and Reboot](#) button will save the field values and restart the MP device using the newly saved values. Note that a system restart is required to effect changes to the network settings and will take around two minutes to complete.

## 4.4 Advanced Set-up

Advanced Set-up utilises access to the Linux operating system on the device and permits full access to all configuration facilities as well as adding and configuring additional software packages.

Access to the device for Advanced Set-up is by means of a command line from a ssh terminal session.

### Connecting to the device

When first flashed with new SECN firmware, the device supports a telnet connection as there is no root password set.

Connect to the MP with telnet using the MP device's Fallback address: **172.31.255.254**

Set PC network to: IP: **172.31.255.253** Netmask: **255.255.255.252**

Alternatively the default br-lan IP address **10.130.1.20** may be used with Netmask **255.255.255.0**

Once the telnet connection has been made, set the root password with the passwd command, logout with the exit command, then reconnect with ssh.

### Setting the device Network Addresses

#### Setting the br-lan Bridge IP Address

Set the unique IP address for the br-lan interface of the MP device by using the uci command, or by directly editing the network configuration file.

From the command line:

```
uci set network.br-lan.ipaddr=103.130.1.xxx    (Where xxx is unique to each MP)
uci commit network
```

Editing the **/etc/config/network** file:

```
config 'interface' 'lan'
    option 'type' 'bridge'
    option 'ifname' 'eth0 bat0 ath1'
    option 'proto' 'static'
    option 'netmask' '255.255.255.0'
    option 'gateway' '10.130.1.1'           # Default router address
    option 'dns' '8.8.8.8'
    option 'ipaddr' '10.130.1.xxx'         # Where xxx is unique to each
device
```

#### Setting the **ath0** IP Address

You may wish to change the **ath0** IP address, however this is not required for basic mesh operation.

From the command line:

```
uci set network.wifi0.ipaddr=10.130.1.xxx    (Where xxx is unique to each MP)
uci commit network
```

Editing the **/etc/config/network** file:

```
config 'interface' 'wifi0'
    option 'ifname' 'ath0'
    option 'proto' 'static'
    option 'ipaddr' '10.10.1.xxx'
    option 'netmask' '255.255.255.0'
    option 'mtu' '1527'
```

### Set the Access Point SSID and WPA Passphrase

From the command line:

```
uci set secn.accesspoint.ssid= VT-SECN-AP
uci set secn.accesspoint.passphrase = potato-potato
uci commit secn
```

Editing the `/etc/config/secn` file: (MP file example)

```
config 'mesh' 'accesspoint'
    option 'wpa_key_mgmt' 'WPA-PSK'
    option 'encryption' 'WPA1'
    option 'ssid' 'VT-SECN-AP'
    option 'passphrase' 'potato-potato'
    option 'ap_enable' '1'
```

**NOTE:** On the MP device running SECN-1.1 firmware, the secn config file parameters are used to automatically generate the hostapd configuration file. Do not edit the hostapd configuration file as it will be overwritten on startup or on use of the web interface.

### Modifying Asterisk Operation

#### Setting up External SIP / VoIP Operation

To add external VoIP support, use the SECN web configuration interface or modify the *secn* configuration file.

From the command line:

```
uci set secn.asterisk.host = sip.myhost.com
uci set secn.asterisk.reghost = sip.myhost.com
uci set secn.asterisk.fromdomain = sip.myhost.com
uci set secn.asterisk.secret = mysecret
uci set secn.asterisk.username = myusername
uci set secn.asterisk.fromusername = myusername
uci commit secn
```

Editing the */etc/config/secn* file:

```
config 'mesh' 'asterisk'
    option 'fromdomain' 'sip.myhost.com'
    option 'host' 'sip.myhost.com'
    option 'reghost' 'sip.myhost.com'
    option 'secret' 'mysecret'
    option 'username' 'myusername'
    option 'fromusername' 'myusername'
    option 'codec1' 'gsm'
    option 'codec2' 'ulaw'
    option 'codec3' 'alaw'
    option 'enablenat' ''
    option 'externip' '0.0.0.0'
    option 'proxy' ''
    option 'softph' 'CLIENT'
    option 'dialout' '#'
    option 'enable' 'checked'
    option 'register' 'checked'
```

#### Dial Plan for SIP / VoIP

The dial plan for external SIP / VoIP operation is defined in the configuration include file */etc/asterisk/potato.extensions.conf* as follows:

```
; Send incoming calls to the MP
exten => s,1,Dial(MP/1)
; Make outgoing calls using [sipaccount] details
; Dial # for access, and then required number string
exten => _#.,1,Dial(SIP/${EXTEN:1}@sipaccount,120,r)
```

## 5. Overview of SECN-1 Operation

This configuration uses Batman-advanced for the mesh rather than Batman as used in earlier firmware versions. Batman-advanced uses a different mesh protocol to batman and so the two will not interoperate on the same mesh.

The MP device provides two physical network interfaces, Ethernet cable and wireless, which are configured as follows:

- The `eth0` interface operates on the MP Ethernet cable connection.
- Two wireless interfaces, `wlan0/ath0` and `wlan0-1/ath0-1`, are set up on the wireless interface `wifi0`.
- Batman-adv is configured to run on the `wlan0-1/ath0-1` interface using the `batctl` command and generates the `bat0` interface.
- The second wireless interface, `wlan0/ath0`, is set up to operate as a WiFi access point.
- The `bat0`, `wlan0/ath0` and `eth0` interfaces are bridged (`br-lan`) together in each MP and assigned a static IP address, and thus, due to the operation of the mesh via `bat0`, all the `ath1` and `eth0` interfaces of all the MPs in the mesh are similarly bridged.
- The default IP address used for the `br-lan` interface is `10.130.1.20`

The mesh will operate in a stand-alone configuration, simply connecting attached devices together and providing telephony between devices. Alternatively the mesh may be interconnected to a LAN to provide access to additional resources, including Internet connectivity.

If one of the devices is connected via Ethernet cable to a LAN router, then all WiFi and Ethernet interfaces connected to the meshed devices will have access to the LAN resources.

If there is a DHCP server running on the LAN (eg in the router/gateway) then devices configured as DHCP clients connected to the mesh node devices via WiFi or Ethernet will acquire an IP address just as if they were connected directly to the LAN.

Note that there is no DHCP server running in a stand-alone mesh arrangement by default, and so in this case, attached devices would need to be statically configured for their IP address in order to connect. Alternatively one of the meshed devices may be configured to provide DHCP service.

### 5.1 IP Address Range for MPs

It should be noted that the IP address used for the `br-lan` bridge in the MP devices needs to be configured during setup, and may or may not be made to lie in the IP address space used on the LAN to which the mesh may be connected. Operation is essentially the same in both cases, but care must be taken to manage the address space in the former case to avoid conflicts with LAN addresses.

IP addresses assigned to MP devices are static. If the IP addresses used for the MP devices lie in the same address space as the LAN, then the DHCP server and other devices on the LAN must be appropriately configured so that the addresses assigned to the MP devices are left free in order to avoid IP address conflicts. In this arrangement, the MP devices will appear on the LAN just as any other device with a static IP address, and they may be accessed for management via browser or ssh terminal session.

Conversely, if the IP address range used for the MP devices is separate to that used on the LAN, the

MP devices will not appear on the LAN and there is no need to reserve the address space. In order to access the MPs for management in this configuration, it is necessary to configure a PC with a static address in the same range as the MPs, and attach via Ethernet cable or WiFi.

The default IP address assigned to the **br-lan** interface in the MPs is **10.130.1.20** which is unlikely to conflict with the default address range of commodity routers.

If it is desired to have the MPs appear on the LAN, the **br-lan** IP address should be assigned accordingly during set up.

The address assigned to the **br-lan** interface for each MP must be changed to be unique, so that each device can provide a separate telephone number. This IP address assignment may be made by a number of methods including telephone IVR, web interface or manipulation of the **/etc/config/network** file.

## 5.2 Batman-Adv Operation

Batman-adv is a "OSI layer 2" routing protocol which is implemented as a kernel module in the Linux kernel. Since Linux 2.6.38 batman-adv is an official part of Linux.

When you assign at least one active physical network interface to batman-advanced, it will create the virtual **bat0** interface. In the SECN-1 firmware **ath0** is assigned to the batman-advanced kernel module. **ath0** is the wireless interface operating in multipoint-to-multipoint mode (ad-hoc).

Because batman-adv operates entirely on MAC layer (OSI layer 2), **wlan0-1/ath0-1** doesn't need any Layer 3 configuration. Only its Layer 2 MAC address is required. The MAC address is configured during production, so we don't need to configure it. All we need to do is make sure to switch **wlan0-1/ath0-1** on. To sum it up: **ath0** is the link-local transport interface for the batman-advanced mesh.

Batman-adv itself bridges all **bat0** interfaces in all the mesh devices to a big, smart, virtual switch. This means that all **bat0** interfaces in the mesh are link-local, even if they are multiple wireless hops away.

Despite being virtual, **bat0** acts like a real, physical, network interface connected to a big switch. As such you can run all kinds of network protocols on it, like IPv4, IPv6, ARP, IPX – or whatever protocol that can communicate over a network interface that is connected link local (which means directly connected, like a straight Ethernet cable connected between two computers, or a bunch of computers connected to a switch).

In the SECN firmware the **bat0** interface itself is again assigned (or rather enslaved) to a bridge in each machine. **bat0** is part of the bridge named **br-lan**, together with **wlan0/ath0** and **eth0**.

**eth0** is the LAN port of the MP, and **wlan0/ath0** is an access-point interface, operating as a Master in WiFi infrastructure mode (as opposed to Client mode used e.g. by a laptop or smartphone)

Hence all **eth0** and **wlan0/ath0** interfaces in all devices running the SECN firmware are part of one big wireless bridge. The **ath0** interface does the low level work to carry the traffic link-locally from hop to hop and batman-advanced takes care about the routes that the MAC packets have to take.

Note: It is not possible to add IP settings to an interface which is encapsulated in a bridge - you can only assign IP settings to the bridge interface itself. **eth0** is part of the bridge **br-lan**, together with **wlan0/ath0**, **bat0** (the batman-adv virtual interface, which is routed by the mesh routing protocol using MAC addresses). Hence you can not assign any IP settings to **eth0**, **wlan0/ath0** or **bat0** - only to **br-lan**.

## BATCTL Command

The following description is taken from the man page published by OpenMesh.org at:

<http://downloads.open-mesh.org/batman/manpages/batctl.8.html>

### Syntax

```
batctl [batctl-options] command [command-options]
```

This command offers a convenient way to configure the batman-adv kernel module as well as displaying debug information such as originator tables, translation tables and the debug log. In combination with a bat-hosts file batctl allows the use of host names instead of MAC addresses.

B.A.T.M.A.N. advanced operates on layer 2. Thus all hosts participating in the virtual switched network are transparently connected together for all protocols above Layer 2. Therefore the common diagnosis tools do not work as expected. To overcome these problems batctl contains the commands ping, traceroute, tcpdump which provide similar functionality to the normal ping(1), traceroute(1), tcpdump(1) commands, but modified to Layer 2 behaviour or using the B.A.T.M.A.N. advanced protocol.

Commands of particular interest include the following:

```
originators|o [-w [interval]] [-n] [-t]
```

Once started batctl will display the list of announced gateways in the network. Use the "-w" option to let batctl refresh the list every second or add a number to let it refresh at a custom interval in seconds (with optional decimal places). If "-n" is given batctl will not replace the MAC addresses with bat-host names in the output. The "-t" option filters all originators that have not been seen for the specified amount of seconds (with optional decimal places) from the output.

```
gw_mode|gw [off|client|server] [sel_class|bandwidth]
```

If no parameter is given the current gateway mode is displayed otherwise the parameter is used to set the gateway mode. The second (optional) argument specifies the selection class (if 'client' was the first argument) or the gateway bandwidth (if 'server' was the first argument). If the node is a server, this parameter is used to inform other nodes in the network about this node's internet connection bandwidth. Just enter any number (optionally followed by "kbit" or "mbit") and the batman-adv module will guess your appropriate gateway class. Use "/" to separate the down- and upload rates. You can omit the upload rate and the module will assume an upload of download / 5.

default: 2000 → gateway class 20

examples: 5000 → gateway class 49

5000kbit

5mbit

5mbit/1024

5mbit/1024kbit

5mbit/1mbit

If the node is a gateway client the parameter will decide which criteria to consider when the batman-adv module has to choose between different internet connections announced by the



aforementioned servers.

### **bat-hosts file**

This file is similar to the `/etc/hosts` file. You can write one MAC address and one host name per line. batctl will search for bat-hosts in `/etc`, your home directory, and the current directory. The found data is used to match MAC address to your provided host name or replace MAC addresses in debug output and logs. Host names are much easier to remember than MAC addresses.

### **Batman-adv and Gateways**

Amongst performance improvements and faster handover of clients, the batman-adv package for the MP now supports configuring advanced batman-adv gateway and gateway client parameters via UCI.

Note: You only need this if you want to use more than one gateway in the mesh. In this case set the gateway MPs mode to Server and the other MPs mode to Client

Gateway settings for Server and Client mode are provided in the SECN web interface on the Advanced page.

Settings may be made from the command line as follows.

An example how to configure batman-adv gateway bandwidth:

```
root@MP-2:/# uci set batman-adv.bat0.gw_mode=server
root@MP-2:/# uci set batman-adv.bat0.gw_bandwidth=384kbit/128kbit
root@MP-2:/# uci commit
root@MP-2:/# /etc/init.d/batman-adv restart
```

Setting the downlink/uplink speed of the gateway like in this example is optional, if you want to override the default value.

For more info check out <http://downloads.open-mesh.org/batman/manpages/batctl.8.html>

An example how to configure a MP as batman-adv gateway client:

```
root@MP-2:/# uci set batman-adv.bat0.gw_mode=client
root@MP-2:/# uci commit
root@MP-2:/# /etc/init.d/batman-adv restart
```

By default the clients will connect to the gateway with the 'best' access (qualified by the TQ route metric, which measures route quality) in the mesh. As long as no other gateway has a TQ route metric which is more than 20 counts better than the currently selected gateway, the clients will stick to the current gateway. If another gateway is more than 20 TQ counts better, the clients will switch the selected gateway. You can, of course, tweak this threshold.

If you want to do some fancy - experimental - setup like dynamically changing the announced gateway bandwidth, in order to balance the load of the gateways, you can change the clients gateway selection algorithm.

Example:

```
root@MP-2:/# uci set batman-adv.bat0.gw_mode=client 1
root@MP-2:/# uci commit
root@MP-2:/# /etc/init.d/batman-adv restart
```

This setting will configure the clients to select the gateway in order to find the best compromise of

TQ route metric and announced gateway speed. A 100Mbit/100Mbit gateway is useless if the TQ route metric is small.

For more detailed info check out the batctl man page at [open-mesh.net](http://open-mesh.net)

## 5.3 Telephony Operation

### Overview

MP devices provide an RJ11 port to which a telephone may be connected and each MP device runs a copy of the Asterisk application to provide the telephony facilities. Asterisk allows phone calls to be made between devices by means of Voice over IP (VoIP) and Session Initiated Protocol (SIP).

### Interactive Voice Response (IVR) Commands

The MP Asterisk configuration includes several telephone extension numbers that allow interaction with the device using Interactive Voice Response (IVR) system. These numbers include:

<b>2661</b>		Read out the mesh wireless interface IP address
<b>2664</b>		Read out the bridge (br-lan, eth0, wifi AP) network interface IP address
<b>7774</b>	RSSI	Read out the rssi signal strength
<b>2426</b>	CHAN	Set wireless channel
<b>2662</b>		Set the unique network IP address of the MP device - Last octet.
<b>2663</b>	CONF	Set the unique network IP address of the MP device - Full IP.
<b>4283</b>	GATE	Set the IP address of the network gateway used by the MP device.
<b>6749</b>	MPGW	Set the mesh batman-adv gateway mode of the MP device.
<b>7466</b>	PINN	Set IVR PIN number. Default pin is 1234
<b>9434</b>	WIFI	Set WiFi passphrase.
<b>3427</b>	DHCP	Enable DHCP temporarily on br-lan to offer Fallback IP.
<b>9322</b>	WEBB	Enable / Disable the web interface
<b>73738</b>	RESET	Restore factory default configuration settings.
<b>9999</b>		Restart Asterisk.

### IVR Command Summary

Commands which change the system configuration require the entry of the IVR PIN number for security. The initial IVR PIN number is set to 1234 This should be changed before deployment.

Commands which have a variable number of input digits will wait for a timeout period after the last digit has been input to complete the command. The command may be terminated immediately by keying in the # digit after the last command digit has been entered.

Commands which require the MP device to be restarted to take effect (e.g. network settings) will include a message advising this fact.

The 2661 and 2662 commands simply read out the IP addresses used by the MP device on the mesh and on the wifi and ethernet network carried on the mesh, respectively.

The RSSI command 7774 will read out the signal strength of neighbouring MP devices. This is intended to assist with adjusting the MP device's location and orientation for best signal from its neighbours on the mesh.

If the mesh has batman-adv gateways set up the RSSI command will list the signal strength values for each neighbour which is selected as a next hop towards a gateway as follows:

Gateway nexthop ... 1 ... 34, Gateway nexthop ... 2 ... 22,

In the absence of gateways the RSSI command will read out the signal strength values for all

neighbours as follows:

Neighbour 1.....36, Neighbour 2.....42, Neighbour 3.....28

Other useful terminal commands for monitoring signal strength are:

<b>wlanconfig ath0-1 list</b>	Lists signal data for nearby devices on the mesh.
<b>wlanconfig ath0 list</b>	Lists signal data for devices attached to the MP's WiFi AP.
<b>batctl o</b>	Lists nearby devices on the mesh.
<b>batctl gw server</b>	Enable batman-adv gateways on the fly

The CHAN command 2426 sets the wireless channel used for the mesh and wifi interfaces.

The CONF command 2663 is used to set the unique network address of the MP device using the full IP number. The 2662 command changes only the last octet of the IP address. If the MP device is attached to a network and the specified IP address is already in use, the commands will fail.

The GATE command 4283 sets the IP address of the network gateway that the MP can use to access IP addresses beyond the local network, such as Internet addresses.

The MPGW command 6749 sets the batman-adv gateway mode of the MP. This setting is used to assist with efficient routing of traffic on the mesh. If more than one MP on the mesh is connected to a gateway, these MP devices should have their gateway mode set to Server, with other MP devices set to Client. If only one MP device on the mesh is connected to a gateway then it is useful to announce this device by setting its gateway mode to Server.

The PINN command 7466 sets the four digit IVR PIN number, which should be changed from the default 1234 before the device is deployed in the field.

The WIFI command 9434 sets the encryption passphrase used for secure wifi access as a numeric string. A minimum of eight characters is required and the passphrase should be changed from the default before field deployment.

The DHCP command 3427 activates a DHCP server temporarily on the device so that if you connect a PC via Ethernet or WiFi it will be automatically given an IP address corresponding to the MP Fallback address. This avoids having to set up a static IP on the PC to connect. The facility can be activated and de-activated with this command, and it is deactivated automatically on a reboot.

The RESET command 73738 restores the device to the original factory default settings.

The Restart Asterisk command 9999 can be useful to ensure that the telephony sub-system is initiated correctly after the mesh network starts up and Internet access becomes available for registering external SIP providers. Restart time for Asterisk is a few seconds, after which dial tone will be available.

## **5.4 Asterisk Operation**

The operation of Asterisk is controlled to a number of configuration files, two of which are of particular interest for MP devices - **/etc/asterisk/extensions.conf** and **/etc/asterisk/sip.conf**

The **extensions.conf** file sets up the dial plan while the **sip.conf** file defines the channels to be used for making calls.

Operation of Asterisk can be monitored from the MP command line by executing the commands:

```
# asterisk -r
# asterisk -vvvvvrddd    Launches with Verbose Lev 5 and Core Debug Lev 3.
```

Some useful commands in the Asterisk shell include:

```
CLI> exit                Return to the command shell
CLI> help                Displays a list of available commands
CLI> core set verbose 5   Set verbose level to 5
CLI> sip reload           Reload sip.conf configuration
CLI> dialplan reload      Reload extensions.conf dialplan
CLI> show dialplan default Display current dial plan
CLI> sip show registry    Display sip registrations
```

## Making Calls to MP Devices

To dial an MP device using the full IP address, dial the IP number substituting the \* character for the dots between octets in the address. To dial an MP with address 10.130.1.21, dial

10\*130\*1\*21

The SECN firmware includes a facility for making on mesh calls using abbreviated dialling by using just the last octet of the MP device's IP address. When an abbreviated number dial string is detected, the full IP address is generated by pre-pending the rest of the address.

The IP address used for on mesh abbreviated dialling is set up during the start up process by the script /bin/generate-extension.sh, using the MP device's own br-lan IP address as reference.

To dial an MP device using abbreviated dialling, simply dial the last octet of the unique IP number assigned to the required MP. This can be dialled as 1, 2 or 3 digits, and may include leading 0 eg

5, 05, 005	(device address 10.130.1.5)
25, 025	(device address 10.130.1.25)
105	(device address 10.130.1.105)

## Debugging Asterisk Operation

Asterisk provides a comprehensive interactive console mode to allow you to monitor its operation.

If Asterisk is already running, invoke the console mode with the command:

```
root@MP-2:/# asterisk -vvvvvrddd
```

This will run the console with Verbosity set to Level 5, and Core Debug set to level 3, which generally gives good visibility of what is happening. It does not interfere with the operation of Asterisk.

An extensive set of commands is available from the Asterisk CLI command line.

To see a list of these commands type: help

To exit the console mode, type: exit

If Asterisk is not already running, you can start it up with console mode running with the command:

```
root@MP-2:/# asterisk -vvvvvrgcddd
```

This can be useful for monitoring the start-up behaviour of Asterisk.

### **Asterisk and Network Settings**

Asterisk has some very particular requirements around network settings, specifically:

#### **Network DNS Address**

Firstly, during Asterisk start-up, it will test for the presence of a ping response from the DNS nameserver address specified in the **/etc/resolv.conf** file. It may wait for a period of many seconds for a response, which will affect the start up delay for the whole device. This delay can be seen in the Asterisk console. In the MP device firmware, the Asterisk start-up script temporarily uses the local host address for the DNS setting to ensure fast start-up.

Secondly, for external SIP / VoIP operation, Asterisk will use the nameserver value in **/etc/resolv.conf** to resolve the URL of the SIP / VoIP host server on the internet. If there is no valid DNS service operating on this address, or the DNS address is not accessible from the MP device, Asterisk will fail to register the SIP / VoIP service and will complain of a DNS error in the Asterisk interactive console output.

#### **Network Gateway Address**

Asterisk requires that the network gateway address specified in **/etc/config/network** be in the same IP subnet range as the MP's IP address, even if there is no device actually present at this address.

If the gateway address is not in the correct subnet, Asterisk will fail to place even on-mesh calls and will complain of a 'Bad file descriptor' error in the interactive console output.

When the MP device's unique IP address is set from the IVR, the Gateway addresses will be set to be in the same IP subnet with the final octet set to '1' e.g. 10.130.1.1

By default, the IVR function will set the DNS address to a public server address at 8.8.8.8

Note: Care must be taken when setting these addresses manually from SECN web interface or command line.

#### **Access to SIP/VoIP Server**

If Asterisk cannot access the network and see the external VoIP host during startup, calls through the service will fail, even if Asterisk is able to register with the service after startup. Calls to mesh devices will work correctly in this scenario, leading to confusion over the status of Asterisk.

This is particularly relevant to MP devices that are connected to the LAN / Internet only via the mesh, as the start up order and timing of scripts in **/etc/rc.d** are designed to ensure the mesh is running correctly before Asterisk tries to start.

#### **Sample Asterisk Console Outputs**

1. Call from MP at **192.168.1.32** to MP at **192.168.1.22** on the mesh network.

```
MP-32*CLI>
-- event_offhook
--   AST_STATE_DOWN:
-- start mp_new
-- event_dtmf 2
-- event_dtmf 2
-- event_digit_timer
--   extension exists, starting PBX 22
-- Executing [22@default:1] Dial("MP/1", "SIP/4000@192.168.1.22") in new stack
```

```
-- Called 4000@192.168.1.22
-- SIP/192.168.1.22-00587578 is ringing
-- Asked to indicate 'Remote end is ringing' condition on channel MP/1
MP-32*CLI>
```

### 2. Call to a PSTN number 0733991234 via SIP / VoIP Service

```
MP-32*CLI>
-- event_offhook
--   AST_STATE_DOWN:
-- start mp_new
-- event_dtmf #
-- event_dtmf 0
-- event_dtmf 7
-- event_dtmf 3
-- event_dtmf 3
-- event_dtmf 9
-- event_dtmf 9
-- event_dtmf 1
-- event_dtmf 2
-- event_dtmf 3
-- event_dtmf 4
-- event_digit_timer
--   extension exists, starting PBX #0733991234
-- Executing [#0733991234@default:1] Dial("MP/1", "SIP/0733991234@sipaccount|120|r")
-- Called 0733991234@sipaccount
-- Asked to indicate 'Remote end is ringing' condition on channel MP/1
MP-32*CLI>
```

## **5.5 Softphone Support**

Softphone Support is provided in order to be able to allow devices such as cell phones and laptop PCs equipped with softphone applications to join the MP telephone network and to make and receive calls on the network, and to an external SIP/VoIP service if configured.

### **Setting up the Devices**

Softphone Support is enabled by the control in the VoIP / SIP section of the Advanced SECN Configuration screen. The available modes are Off (default), Master and Client.

In order to support Softphones on a network over the mesh, one, and one only, device on the network is set to Master mode. The copy of Asterisk running on the Master device is used to route softphone calls around the network.

The Master device will automatically have its IP address last octet set to [.252](#)

This address is reserved by default in a SECN network as a 'well known' network address for the Softphone server.

Other MP devices on the network that are to be able to make calls to softphone equipped devices must have their Softphone Support control set to Client.

Note that after setting the mode in the configuration screen, the device has to be restarted for the changes to take effect.

### **Configuration of Softphone Accounts**

Softphone accounts are defined in the file `/etc/asterisk/softphone.sip.conf`

By default there are ten accounts set up for softphones defined as softph300 through softph309

Once assigned to particular attached softphone devices, these devices may be called using their three digit numbers 300 through 309.

The list of softphone accounts may be extended as required, and the individual passwords changed as required by manually editing the configuration file.

Note that setting of the allowed codec(s) is critical to the operation of some softphone clients.

It has been found for example that SipDroid will operate correctly only when ulaw is the only allowed codec.

A section of the `/etc/asterisk/softphone.sip.conf` file is shown below for reference.

```
[softph300]
type=friend
secret=Pa55uu0rd300
context=default
host=dynamic
disallow=all
;allow=gsm
allow=ulaw
;allow=alaw
dtmfmode=rfc2833
qualify=yes
canreinvite=no
nat=yes
```

### Setting up the DHCP Server

Telephony on the SECN-1 MP network relies on the IP addresses of the devices attached to the network to route calls to the correct device. MP devices typically have statically assigned IP addresses for this purpose. This allows a MP network to operate without the need for any master device controlling the telephony system, thus providing maximum robustness.

This is not the case with Softphone Support described here. Softphone devices are 'registered' with the Softphone Master device, and the presence and correct operation of this device is essential for softphone operation. It is a single point of failure.

Furthermore, the softphones do not rely on their IP address to determine their phone number; the phone number is part of the registered account for the device.

However a device which attaches to the network may not have a static IP assigned and will expect to get an IP address from a DHCP server on the network. When a cell phone or similar device equipped with a softphone application is attached to the network it is generally configured to receive an IP address from a DHCP server.

Where a MP network is attached to a LAN, there will usually be some device on the LAN running a DHCP server that will hand out a suitable IP address to an attaching device. As long as the MP static addresses are on the same sub net range as the DHCP addresses, all will be well.

Where a MP network is operating in a stand alone manner, not attached to a LAN, there will be no device present to hand out IP addresses. For this reason, a DHCP Server is provided in the firmware so that an MP device can perform this function.

The DHCP Server may be configured from the DHCP Server section of the Advanced SECN Configuration web page.

Care should be taken to avoid IP address conflicts, and conflicts between multiple DHCP servers on the same network. The range of addresses used for the DHCP server should be outside the range used for statically assigned addresses used for the MP devices.

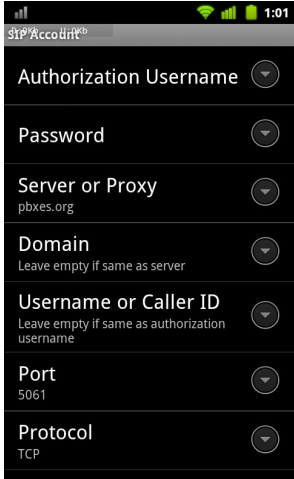


### Setting up the Softphone Clients

For a Sipdroid client, the setup is as follows:

1. Start up Sipdroid and go to the Sipdroid settings.
2. Create a SIP account with Authorization Username set to one of the account entries in the file **softphone.sip.conf** (e.g. softph300),
3. Set the Password to match the account entry e.g. "Pa55uu0rd300"
4. Set the Server (or Proxy) to the IP address of the Softphone Master MP (ie .252 on the sub net)

Sipdroid should show successful registration to the softphone server.



The screenshot shows the Sipdroid settings interface. At the top, it says 'SIP Account'. Below are several settings, each with a dropdown arrow on the right: 'Authorization Username', 'Password', 'Server or Proxy' (with 'pbxes.org' visible), 'Domain' (with 'Leave empty if same as server' below it), 'Username or Caller ID' (with 'Leave empty if same as authorization username' below it), 'Port' (with '5061' visible), and 'Protocol' (with 'TCP' visible).

### Making Calls to and from Softphones

To make a call from a softphone equipped device to an MP device simply dial the last octet of the MP's IP number in the usual manner.

To make a call from an MP device to a softphone device, simply dial the three digit number corresponding to the account entry e.g. "300"

Calls to softphone clients are only supported from MP devices with Softphone support set to "Client", and from the softphone Master device.

## 5.6 USB Extended File System

The SECN-1.1 firmware supports additional USB flash memory storage on devices that are equipped with USB ports. Examples of these devices include the TP-Link WR703N, MR3020, MR11U and WR842ND devices for which SECN-1.1 firmware has been ported.

USB drives on these are automounted to **/mnt** as normal unless they are labelled with one of two special volume names: SECN-Extended and WEBSITES. These labels enable two special purpose USB configurations which are used to support additional installed program packages and local web server content for the SECN-1.1 firmware.

### Extended filesystem for additional packages

The first pre-defined USB configuration requires the USB drive to be formatted as ext3 and have a volume label of "SECN-Extended". Formatted and labeled this way, the USB drive will be automounted to **/user** instead of **/mnt**

There is a file "**SECN-extended.tgz**" available with the firmware which contains an extended filesystem for the device, including a pre-installed copy of Asterisk configured for use with SECN to support telephony in the same way as the MP-01 devices, including softphone support, but without the built-in ATA.

To set up a USB memory for this configuration, format the USB as ext3 e.g.

```
$ mkfs.ext3 /dev/sda1
```

then label the drive "SECN-Extended" so that it gets automounted under **/user** e.g.

```
$ e2label /dev/sda1 SECN-Extended
```

Note: PLEASE be sure that you run mkfs only on your intended USB drive.

A good way to check is to run:

```
$ cat /proc/partitions
```

and verify the drive's device node.

Alternatively you may use an application such as Gparted to format and label the USB device.

After formatting and labelling the USB flash drive, unpack the "**SECN-extended.tgz**" file into the root of the drive. The extended filesystem drive is now ready for use on the TP-Link SECN 1.1 device. With the TP-Link device turned off, insert the USB flash drive, and power up. If you have followed the steps correctly, the USB drive will be automounted to **/user**.

Simply enter the mount command to verify.

Note: As of this writing, due to a bug in the OpenWRT automount feature, inserting the above drive in the TP-Link device while it is running will result in it being mounted to **/mnt** instead of **/user**. This won't hurt anything, but until you reboot, the features available on the extended filesystem won't be available due to the incorrect mount point.

### Installing additional packages

Other packages may be installed into this flash memory space with the command:

```
root@MP-2:/# opkg install -d usb <package-name>
```

### Installing web content

There is a directory called **/websites** on this USB ext3 file system which may be used to store web content.

This directory appears as **/user/websites** and is symlinked to **/www/websites** on the device, so the content may be accessed through the web server at:

`http://<ip-address>/websites`

### Using a VFAT USB for web content from Windows

The second pre-defined USB configuration is formatted as the normal FAT32 (vfat) file system and has a volume label of "WEBSITES". This was done to more easily allow Windows users to capture websites to a USB drive since Windows support of the ext3 formatted drive is limited.

This volume is mapped to **/www/websites2** and so the web content will appear at:

`http://<ip-address>/websites2`

If you want to simply capture websites on a FAT32 USB drive (vfat), give it a volume label of "WEBSITES" and it will automount at boot up to **/www/websites2**.

To capture websites on a USB drive under either Linux or Windows, a good free utility is

HTTrack Website Copier.

There are both Linux and Windows versions at: <http://www.httrack.com>

END OF DOCUMENT