Policy Negotiation in Open Mobile Networks

Chase Miller, Maxim Kovalev, Nandita Joshi and Rishik Dhar
ECE, CMU SV
{chase.miller,maxim.kovalev,nandita.joshi,rishik.dhar}@sv.cmu.edu
Advisor
Patrick Tague
ECE, CMU SV
patrick.tague@sv.cmu.edu

***Abstract*—Next generation wireless communication networks, such as Carnegie Mellon's CROSSMobile architecture, aim to increase network performance by enabling the exchange of information across different layers of the network in a secure manner. Accomplishing this requires a framework for Shared Infrastructure or Infrastructure as a Service for Cellular Network Services with configurable and adaptive policies for the fair provisioning and utilization of resources. The architecture envisions an open marketplace environment where various components in the network can negotiate and share on-demand services. This paper covers the various aspects of developing a self-organizing, trust based policy framework for CROSSMobile. Our goal in this project was to propose a protocol that governs how participants in an open network may establish trust, explore available resources or services, and transact on these resources and services so the network can behave like an ecosystem, which can sustain on its own and provide rich communication services to consumers/devices connected to it. The driving principle for the protocol design was human-to-human social networks where information about availability, quality, and reliability of services and resources are shared through word of mouth and recommendations are based on personal experiences. We then elaborate on the simulation framework and explore possible security threats, challenges, and future work.**

I. Introduction

Our work builds on top of a couple of previous studies and course projects by the students of CMU Silicon Valley, CMU's Cylab, and HP Laboratories. The CROSSMobile network architecture is designed to allow open interaction between different network components. Opening the market in terms of network resources and services provides enticing opportunities for customization and opens up previously unexplored risks in terms of establishing, maintaining and monitoring safe transactions between vendors with vested interests that need to depend on each other for conducting business. Since the nodes in the network could be any third party, trust and security arise as major concerns while negotiating. The original idea that was tried out by our predecessors was a prototype of a standalone simulator with all the network behavior implemented as a part of the simulation. We have taken a step forward by breaking down the behavior into different classes that can be inherited or composed into an aggregate class which may choose to exhibit different behaviors in different situations. To ensure conflict free interactions, we envision the creation of a collaborative policy agreement mechanism that allows the different parties involved to provide a list of their required and desired features based on which they reach a mutual agreement factoring the least common requirements and greatest common desirability.

II. Related Works and Motivation

As an example of Open Network we studied the CROSSMobile Cross-Layer Architecture as defined in [1]. In particular, we concentrate on certain challenges for participants in open networks with wireless devices and the apps that use the network.

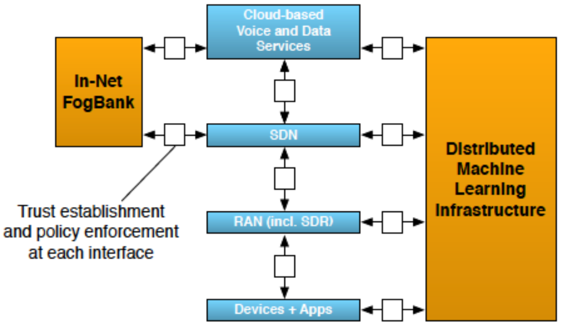
Cloud-based Voice and Data Services
In-Net FogBank
SDN
Distributed Machine Learning Infrastructure
Trust establishment and policy enforcement at each interface
RAN (incl. SDR)
Devices + Apps

Fig. 1: CROSSMobile system architecture, combining software defined networking (SDN), radio access network (RAN) utilizing software defined radio (SDR) techniques for frequency and air interface agility, throughout-the-network fog computing capability, and a set of open interfaces for policy-based cross-layer trust establishment and state information sharing. Policy-based interfaces support the creation of a network out of Mutually-Suspicious Elements (MSEs)

A. CROSSMobile System

The wireless ecosystem (encompassing present-day cellular and non-cellular networks) is made up of many different players see Figure 1, each with issues and opportunities unique to their role. A rational approach to the problems faced by the players must make sense in the context of each of these players. We borrow the definitions directly from [1] and mention them here as the motivation for our project.

1) **Operators** of unlicensed wireless networks enterprises, small businesses and homeowners lack the means to

pool their individual investments in WiFi (and other) networks even when it makes economic sense to do so.

2) **Application Service Providers** (ASPs) and App Developers face a nearly completely opaque network that stands between them and their customers, yet the network as time-varying characteristics (especially at the RAN level as defined in [1]) have a first-order effect on service quality.
3) **End Users** regularly experience variable network performance, manifested as call drops and apps that hang on high-latency or failed network transactions. This experience is the product of the contributions of Telecommunications equipment manufacturers (TEMs), carriers and ASPs with the user left as the system integrator.

B. *Policy Based Negotiation*

Negotiation is a part of our daily lives, be it serious applications such as in economics, game theory, and management, or mundane transactions such as plumbing or gardening contracts. It is in human nature to negotiate best deals, and our approach in this project was to establish a negotiation strategy which closely imitates human behavior. Our goal was to propose and develop a protocol that embodies the negotiation strategy that closely resembles the art and science of negotiation as practiced by us in our day-to-day activities.

In our survey we came across several studies that emphasize the need of policy based protocols, one such study [3] suggests that, "... a protocol to be used by each user in the network is needed to fulfill the requirement for both node and its neighboring node. This protocol should provide the information for each user to satisfy the provided services from each other and also provide the allocation of limited resources wisely by using user policy. The node that provided services, checks requested attributes and values, with its policy and replies result, accept or reject or re-negotiate, to its neighbor node."

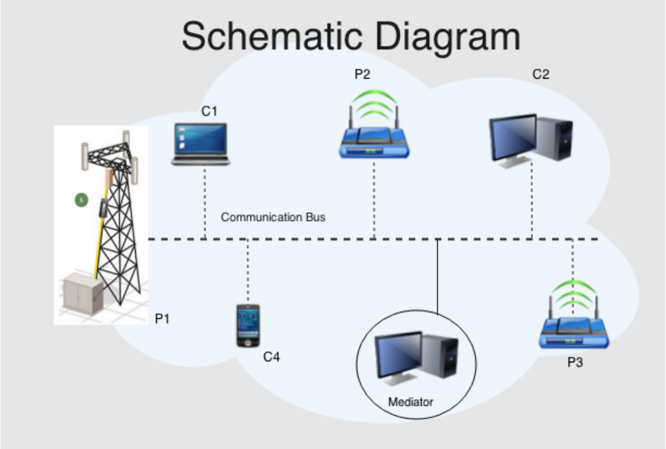

Fig. 2: Schematic Representation of an Ad-Hoc Network using Policy Based Negotiation Protocol to share services and resources across.

III. Protocol Design

The authors in [2] put forth a very clear definition of a protocol. The protocol determines the flow of messages between the negotiating parties, dictating who can say what and when. It also acts as the rules by which the negotiating parties must abide if they are to interact. Moreover, the protocols provide the participants with trust guarantees, that no party has access to extra information or is able to forge false information. Its value to negotiation hosts such as auction houses and market makers is that it provides a standard framework that all potential customers can use to interact with them. However, it does not require a specific market mechanism, so it allows the host to decide on an appropriate one.

A. *Conceptual Design*

Our predecessors started the exploration with a two-layer architecture, see Figure 3, which used a very simplified peer-to-peer trust based protocol. One of the limiting factors of the simple design was that participants did not have a well defined behavior. Additionally, there was no easy way to distinguish the simulation from the components that made it up. Even with these limitations, it served as a great prototype. During the early stages of our project, we were able to leverage many of the preexisting ideas to work towards building the concept further into a three-layered architecture as explained in the following subsections.

1) Two Layer Architecture: In this system, the two layers are Providers and Consumers. Their corresponding behaviors (high-level) are defined below.

- Provider packets:
  - Advertise itself: price, bandwidth, scope
  - Reject demand temporarily
  - Reject demand permanently
  - Offer unilaterally signed contract according to the demand
  - Send access tokens - the equivalent of service
- Consumer packets:
  - Send demand to an entity advertising desirable resource
  - Sign unilaterally signed contract
  - Provider talks to many consumers
  - Consumer talks to one (any satisfactory) provider

2) Three Layer Architecture: Figure 5 explains the idea of three layered architecture and how the third layer may participate in the negotiations. The core idea is that we should rely on the human like model where a random contact in the network can be leveraged to make recommendations, serve as mediator, and provide reputation validation service, so that the participants in a transaction have a third-party helping reach agreements and provide reconciliations. Given the transitory nature of participants in a network, we recommend a new approach, we call network-persistent data. This information resides on the network where all participants have certain pieces of information based on their history and what they have observed since joining the network.

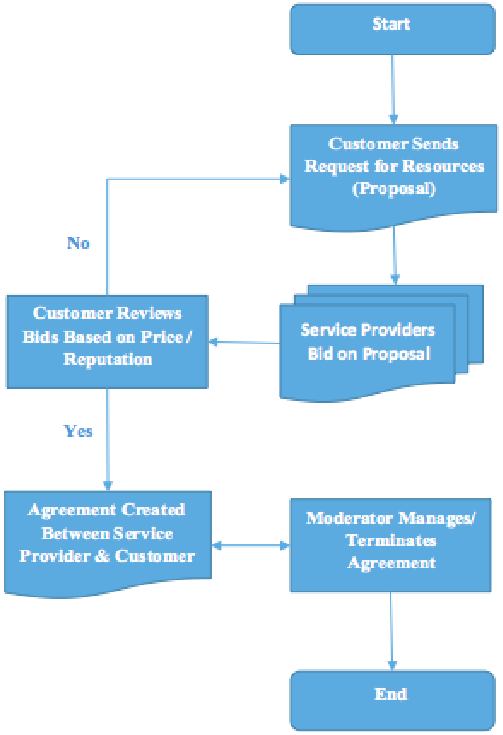
Start
Customer Sends Request for Resources (Proposal)
No
Customer Reviews Bids Based on Price / Reputation
Service Providers Bid on Proposal
Yes
Agreement Created Between Service Provider & Customer
Moderator Manages/ Terminates Agreement
End

Fig. 3: Flow chart explaining the Flow of communication in Two Layered Architecture

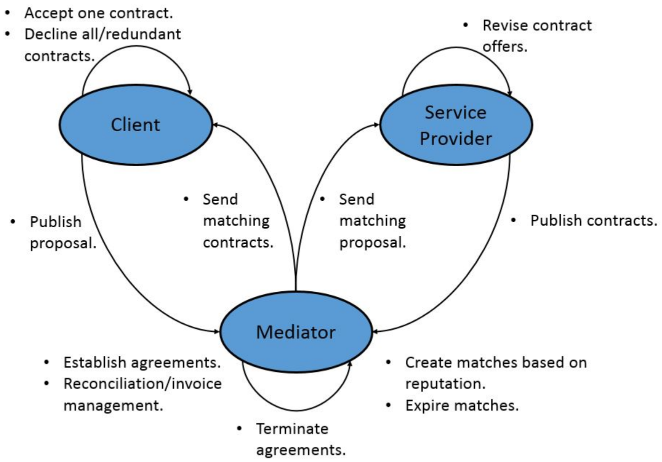
• Accept one contract.
• Decline all/redundant contracts.
• Revise contract offers.
Client
Service Provider
• Publish proposal.
• Send matching contracts.
• Send matching proposal.
• Publish contracts.
Mediator
• Establish agreements.
• Reconciliation/invoice management.
• Create matches based on reputation.
• Expire matches.
• Terminate agreements.

Fig. 4: Schematic representation of the Flow of Communication in Three Layered Architecture

B. Software Design and Implementation

We have created behavioral abstractions and the implementation closely matches the defined abstractions. We designed the framework to be both lightweight and highly extensible. Looking at the code https://github.com/maxikov/wirelessnegotiator should be enough to understand how the current version has been implemented. We define the behavioral components as follows, Design (python Implementation):

- Simulator (policyNegotiationSim.py)
- Consumer (Consumer.py)
- Provider (Provider.py)
- Communication Bus (CommunicationBus.py)
- Entity (Entity.py)
- Contract (Contract.py)
- Proposal (Proposal.py)
- Mediator - *Implementation Pending*
- Repuation - *Implementation Pending*

C. Running the Simulation

The framework git repository can be cloned from https://github.com/maxikov/wirelessnegotiator. Reference the Software Design and Implementation section to better understand the code structure. Running the following command from the cloned repository directory will start the simulation.

```
python policyNegotiationSim.py
```

The Policy Negotiation Simulation initializes the network with Consumers and Providers that interact over a common Communication Bus. Consumers send out Proposals for Entities including desired price range and reputation level. Providers send out Contracts for Entities offered at a certain price range based on reputation of the provider and the current demand. Mediator matches Proposals with Contracts, helps Providers and Consumers reach agreement, mediates:

- Exchange of resource with invoice,
- Service quality and payments and
- Reputation of Providers and Consumers.

D. Security Threats Simulation

The current simulation model is extensible and abstract enough to simulate security threats mentioned in the Findings and Recommendations sections. As the implementation of security threats depends on the mediator, and the implementation of a reputation mechanism, this has been defined under future work for now.

IV. Findings and Recommendations

Though designing and running the negotiation simulation, we have identified key findings. Based on those findings, we have created several recommendations for moving forward with this protocol design.

A. Notable Findings

In the protocol's current state, a race condition occurs when multiple clients respond to the same resource advertisement from an access point. In this simulation, if the access point attempts to serve two clients simultaneously, it could result in a shortage of resources, denying service to all clients being served by the access point. This race condition could be avoided through the introduction of a rejection mechanism. This mechanism would allow the access point to serve only one client per resource advertisement, rejecting the other demands to ensure that sufficient resources are always available.

In addition to the rejection mechanism, there is a need for a fairness mechanism to avoid the issue of new clients appearing and being served before other, older clients have been served. This mechanism will ensure that all clients are served in a fair and timely manner. This fairness mechanism could also include situational configurations. For example, in the event of an emergency, emergency responders would move to the front of the queue and receive whatever resources they need.

B. Possible Threats and Malicious Behavior

Finally, we have identified a variety of ways that the current model can be exploited. This section elaborates on some of the attacks and their mitigations.

1) Denial of service via resource hoarding:

- Consider the case of a malicious service provider. A malicious service provider could advertise that they have resources that they do not actually have. Without proof of resources, a client would sign the contract, believing that the resources are available when they're really not.
- Similar to the scenario above in terms of impact on the network, there could be a malicious client in the system. A malicious client could request resources with no intention of paying for them. This ties up resources that could otherwise be offered to others.

Mitigating such attacks requires a mechanism of authentication and authorization of entities in the network. For the malicious service providers, a resource availability verification system could be introduced. This could either be handled by a third-party or made an organic part of the system. To mitigate the case of resource hoarding by malicious clients, the client could be made to provide an upfront payment for all or a fraction of the resources requested. Additionally, a timeout period can be introduced within which the clients would be expected to complete their request and transaction.

2) Distributed Denial of Service via resource hoarding:

- While similar in nature to the denial of service attempts explained above, this attack would be harder to detect in the system. Its main concept is that there could be multiple malicious service providers present in the system who collude to advertise that they have resources that they do not actually have. Again, a client would sign the contract without proof of resources, believing that the resources are available when they?re really not.
- Likewise, multiple malicious clients could request small quantities of resources each (summing up to a large fraction of resources available) with no intention of paying for them. This ties up resources that could otherwise be offered to others.

Mitigating these attacks is not as simple as forcing the malicious service providers to provide details of their resource availability or getting clients to make upfront payments. As multiple service providers and multiple clients collude, the individual stakes are much lower, while producing a massive impact on the open network. Such cases can be mitigated to an extent with the introduction of a reputation mechanism in the network.

3) False Reputation Update via spoofing service providers/clients:

- A service provider or client can spoof the identity of any other service provider or client and misuse network resources under this false identity. This would affect the reputation of the target service provider/client in the network. One way to mitigate such a scenario is introducing authentication of entities when they join the network. A signature based scheme along the lines of the
- Public Key Infrastructure model could be added to the network.

4) Data Leakage:

- As the open mobile network consists of a nascent framework with some information owned by service providers and clients that they do want to make public and some other information that they want to be protected, attempts of man-in-the-middle attacks on the transmitted data to either obtain or modify it would occur and need to be considered.
- Data encryption and authentication should be implemented for the information transmitted in the network to mitigate such attempts.

5) False Reputation Report: In the network we have implemented, entities keep and exchange reputation information about third parties directly between each other. An adversary can give false information in this case and affect the reputation of entities in the network. For example, If Entity A says that entity B had poor reputation, Entity B can tell everyone that entity A is not trustworthy. To mitigate such reputation manipulation attacks, the moderator in the network can be assigned the task of monitoring the reputation database, propagating its information in the network, and keeping it updated. This of course assumes that the moderator is verified in some manner and is a trusted entity.

C. Recommendations

The framework that we have created is a communication-based framework for policy negotiation. Currently, it supports local area networks (LANs) via a single, common communication bus. In the future, we envision a decentralized protocol that can be implemented into wide area networks (WANs) with multiple buses and bridges. Additionally, when creating this protocol we intentionally left it modular and generic enough so that it can easily be modified, experimented with, and scaled to fit different sized networks.

V. Conclusion and Future Work

A large portion of our work was understanding and redefining certain ideas taken from our predecessors and other research work done in the area Policy Based Negotiation. We were able to simplify the design of the framework to encourage easy extension and experimentation on the ideas that have so far remained mostly theoretical. The current framework can be extended to simulate the behavior of participants in an Open Network, subsequently the negotiation policies that are developed can be tested on such networks. CROSSMobile is the most attractive test bed for these protocols as it is a live system with minimal cost and high degree of support for tuning and optimization iterations.

As future work we propose breaking down the exploration into several different categories.

1) Further Framework Implementation: The implementation of a Mediator and Reputation Service that uses network-persistent data about the reputation of participants. This includes implementing the use of Reputation and Price for negotiation on proposals and contracts.

2) Threat Simulation: There is tremendous opportunity to simulate security threats as identified in the Findings and Recommendations section above.

3) Negotiation and Profit Maximization Models: Define economics based models and techniques such as Nash-Equilibrium for optimizing the exchange of resources and services in a network.

4) Policy Definition and Implementation: There is also scope to define some utility functions that can be used as the objective functions for optimization by the participants. There is a good opportunity to research the inclusion of Emergency Service Requests as a major change agent in how a network may dynamically chose a different policy for a certain time period (identified as emergency period).

Acknowledgment

Thanks to Prof. Bob Iannucci, Prof. Patrick Tague and Brian Ricks for providing clear direction on what needed to be done and what parts to focus on. Their valuable guidance and suggestions allowed us to reach deeper understanding of the key concepts used to build the prototype.

References

[1] Bob Iannucci, Patrick Tague, Ole Mengshoel, and Jason Lohn, *CROSS-Mobile: A Cross-Layer Architecture for Next-Generation Wireless Systems*

[2] Claudio Bartolini, Chris Preist, *A Framework for Automated Negotiation*

[3] Srifa, N. and Pornavalai, C. and Varakulsiripunth, R., *A Policy Based Negotiation Protocol for Services Agreement in Mobile*,in Information, Communications and Signal Processing, 2005 Fifth International Conference, 2005

[4] Mridula Shastry, Nandita Joshi, Sakshi Goel and Tushar Dadlani *Policy Negotiation for Open Mobile Networks*, Submitted as a part of Course Work Term Project Report, in Fall 2014, Wireless Networks, CMU SV

[5] Rajiv Maheswaran and Tamer Basar *On Revenue Generation When Aucitoning Network Resources*, in Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005.

Fig. 5: Watch a short video summarizing this report. YouTube Link: https://www.youtube.com/watch?v=7Y3O6vQGYNgfeature=youtu.be