Abstract—There is little existing work which empirically analyses network management practice with respect to policy creation and maintenance. We carried out five semi-structured interviews of network administrators at a diverse range of enterprises. Analysis of the results led us to make the following contributions: Provide a description of the context in which network administrators carry out their daily work; we introduce a number of policy space dimensions; we identify a set of real-world network policies, and show how each of these may be represented by these dimensions. There are a number of use cases for our work: Engineers or researchers looking to create or refine high-level network management frameworks; Network administrators looking to document their network policy in a consistent format; Researchers looking for empirical research to compare their own work to.

I. INTRODUCTION

While reviewing the relevant literature we found little work which empirically analyses network management practice with respect to policy creation and maintenance. Furthermore, we found that there is a need for standardised representations of network policy, and that network programming languages lack empirically-validated motivations.

In this report we present a study we undertook in which we carried out five semi-structured interviews of network administrators at a diverse range of enterprises. The goal of this study was to determine the concepts needed to describe enterprise network policies (we call these the ‘dimensions’ of policy space’).

We make a number of contributions. Specifically: We provide a description of the context in which network administrators carry out their daily work; we introduce a number of policy space dimensions; we identify a set of real-world network policies, and show how each of these may be represented by these dimensions.

We envision a number of use cases for our work: Engineers or researchers looking to create or refine high-level network management frameworks; Network administrators looking to document their network policy in a consistent format; Researchers looking for empirical research to compare their own work to.

A. Problem Statement

The goal of this study is to determine the concepts needed to describe enterprise network policies. We refer to these concepts as the ‘dimensions’ of policy space; that is, orthogonal, non-overlapping features which can be used to express policies.

This problem led us to ask the following exploratory questions:

- Who sets network policy?
- What sorts of things do network policies restrict?
- Can policies be classified or categorised in some manner?
- Why are these policies created?
- Are there problems with existing workflows for creating network policies?

We refined these into concrete research questions, which are discussed in greater detail below.

B. Research Questions

1) RQ1: What do network administrators restrict in enterprise networks?: Motivation: We will collect descriptions of network policies from network administrators. By analysing these for commonalities we hope to identify some of the dimensions of policy space (or at least make inferences which will allow us to introduce hypothetical dimensions whose existence can be tested).

2) RQ2: What are the drivers of policy dynamism?: Motivation: Identifying the drivers which cause restrictions to be created and modified will provide context and enable us to refine our findings from RQ1. From the identified drivers we may also be able to derive additional examples of network restrictions, which can in turn be used to derive policy dimensions. For example:

- Not all dimensions identified in RQ1 may be necessary for a generalised model of policy space like we seek to develop. Eg: They could be highly specialised for a particular network.
- There might be some additional dimensions beyond those identified in answering RQ1. Eg: There might be a more efficient way of implementing some policies by introducing new dimensions.
- Understanding what causes policies to be modified will hopefully allow us to identify a set of entities which facilitate the creation of policies which do not need to be modified as often, or which can be more efficiently or easily modified. Note that this assumes that modifying policies has some cost which we want to avoid paying if we can.
3) **RQ3: Does policy dynamism affect network restrictions?**

Motivation: Change in the policy population may influence the kinds of restrictions administrators create. For example:

- There might be more change along some dimensions of policy space than others (eg: ‘user’ components are frequently updated, but ‘time’ components are not).
  - If existing tools for creating restrictions are not expressive enough this might bias admins towards creating time-based restrictions rather than user-based ones because the former require less work to create.
  - Admins might be biased towards creating time-based restrictions rather than user-based ones because the former require less work to maintain.
- Changes along some dimensions might be more expensive than others (in terms of network admin workload, CPU cycles, risk, etc).
- Policy drivers may be correlated to dimensions (eg: some of the drivers identified in RQ2 might tend to cause changes along the same dimensions).

4) **RQ4: How are policies implemented in industry?**

Motivation: Different mechanisms (e.g. software scripts, managerial edicts, physical changes etc.) and processes/workflows might be better suited to addressing some entities more than others. This could bias responses to the other research questions towards some entities and away from others and should be taken into account.

E.g. Manual configuration or automated scripts might make it easy to implement policies which target specific devices, but difficult to target particular classes of device, or traffic from particular users. ie: Because of the difficulty of addressing some entities network administrators may simply not bother (and it is conceivable that they may not have put much thought into why this is, and so we should investigate).

II. RELATED WORK

In this section we present a brief review of the relevant literature. This review had three main findings, which we elaborate on below.

- There is a paucity of empirical network management research.
- There is a need for standardised representations of network policy.
- Network programming languages lack empirically-validated motivations.

A. Paucity of empirical network management research

Our review of the literature revealed only a few studies which empirically investigate network administrators’ daily work.

Kraemer [1] interviewed eight network administrators (adopting an exploratory qualitative research approach based on semi-structured interviews, similar to our own). However, the participants all worked at the same institution, whereas our participants were all drawn from different organisations (all of which operate in different domains). Kraemer’s focus (the causes of human error in network and information security management) is also quite different from ours.

Kim [2] surveyed several hundred network operators, but focused on their impressions of Kinetic (a framework developed by Kim which provides abstractions for network management), whereas we are coming from the opposite direction: Trying to identify abstractions which could be adopted by a network management framework. Kim’s survey is also quite limited in some respects: Participants were self-selected, the survey questions were highly specific to the Kinetic platform, little information was given about the sampled population, no formal study design was presented, and the paper does not present a discussion of the validity of the study.

Bhattacherjee [3] interviewed two information technology (IT) workers as part of a case study investigating the link between business processes and information technology. While the authors discussed some IT administration processes (for example: the way in which users were able to access corporate data, and the way in which user-developed software could be deployed), the primary goal of this study (to investigate the role of IT in meeting the business needs of organisations) was quite different to ours (to present a generalised set of concepts for expressing enterprise network policies). The study is also over 20 years old at the time of writing, potentially limiting its relevance today.

B. A need for high-level, standardised representations of network policy

Our review of the literature leads us to believe that there is a need for high-level and standardised approaches to defining and implementing network policy.

Kraemer [1] finds that network administrators feel that the technologies they use to maintain network security are insufficient, and wish that there were well defined and standardised procedures for defining and implementing network policy. Kraemer also notes that administrators sometimes introduce security policy violations themselves, in order to support certain user use cases, or to save time (corner cutting). This suggests to us that the management technologies being used are not flexible enough. Kim’s [2] survey of 870 experienced network operators found that most were not confident that they could reconfigure a network without (at least initially) introducing bugs, and that they desired tools for automating and error-checking (verifying) reconfiguration tasks.

Fulford [4] states that there is a “high degree of variety in terms of [the] content and dissemination [of information security policies]”. In particular, Fulford notes that one of the primary challenges of information security management is communicating the policy to employees (IT users), and
to institutions themselves. Kraemer [1] found that poor inter-administrator communication could lead to inconsistent application of network policy. We argue that a model (such as the one we propose in this report) for formally expressing network policy will help alleviate issues related to the communication and interpretation of network policy.

However, standardised representations of network policy alone are not enough. The literature generally agrees that the task of network management is becoming increasingly complex and that new approaches are needed to address this challenge. [5]–[9] One approach which has been suggested is to raise the level of abstraction at which network management is carried out [11], and this is the same general. In the area of software-defined networking (SDN) there is a growing body of work around the idea of network programming [2], [12]–[15]. However, such work is still quite low-level, requiring most policies to operate at the level of individual packets or packet flows.

C. Network programming languages lack empirically-validated motivations

There are a number of papers which introduce and/or survey languages for representing network policies [6]–[8], [10], [15]–[24], but little work has been presented to empirically validate these languages (eg: by assessing their usability in practice, or by demonstrating a clear link between their core abstractions and the needs of network administrators). Strassner [25] presents a language specifically designed for mapping high level business requirements to the network configurations required to satisfy them, however the RFC is incomplete and details about the grammar of the language are unavailable. Casado [26] recommends that SDNs support a number of high-level features, but does not provide empirical evidence demonstrating why these features, in particular, are compelling. Trois [27] builds on Casado’s work to present a taxonomy of SDN programming languages, but again (to our knowledge) there is little empirical evidence to motivate the features supported by these languages.

D. Other findings

We also note a number of minor points from the literature. Authors occasionally make statements about network management without empirical evidence. While their claims are often plausible, and may well be true, they should be validated. For example, [12] says that “networks operate in a constant state of flux”, and that “configuration changes are a common source of instability in networks”, but does not provide evidence for these statements.

Network policy is influenced by a number of factors, not all of which are germane to the network’s main purpose. [1] In our study we identify a number of these factors as part of an effort to understand the context in which network policy is created and implemented.

There are different understandings of what a ‘policy’ is, even within the context of information technology. [17] For example, the policy-based network management (PBNM) literature views ‘policies’ as concise, atomic descriptions of intended network behaviour, and aims to introduce standardised concepts for expressing them, while Fulford [4] uses ‘policy’ to refer to a single, monolithic document intended for humans to read.

Kraemer [1] notes that network administrators have a high workload. Our interviews with network administrators supports this finding, and we additionally find that the participants of our study all worked in very small teams and consequently had wide-ranging responsibilities.

III. Methodology

We determined that survey research was the method best suited to our situation. It involves the selection of a representative sample from a well-defined population, and the use of data analysis techniques to generalize from that sample to the chosen population. It is used to identify the characteristics of a broad population, and requires a clear research question that asks about the nature of a particular target population. [28]

In particular, we carried out five semi-structured interviews of network administrators for this study, each lasting between 40 and 60 minutes. This exploratory qualitative research method falls under the broad category of survey research. It is suited to identifying the broad characteristics of a target population. [28] In the following subsections we describe the process we followed to develop our research methodology.

The first two of these interviews were meant to be part of a pilot study, intended to refine the main study. However, because the response rate rate for the study was very low we felt that the data from the pilot was too valuable to exclude from the main study. We also felt that, while some refinement took place after the pilot, the execution of the interviews pre and post-pilot was sufficiently similar to justify the inclusion of this data.

The study was conducted over a period of two months, from when data collection began in the pilot phase until analysis ended. All data collection and analysis was carried out by a single researcher, with supervision from two senior colleagues.

A. Data Collection Procedures

1) Technique: For our data collection technique we chose semi-structured interviews. These involve an interviewer (who in our case was also the principal researcher) and a participant discussing a number of talking points. They have a conversational flow, centred on broad questions. Interviewers have the freedom to introduce new questions and adapt existing ones in order to probe unexpected results. [29] This makes them well suited to exploratory studies which aim to generate new theories, rather than confirm existing ones.

We conducted one interview per study participant and held all but one interview at each participant’s place of work. One
interview was conducted over the phone as we were not able to travel to the participant’s location.

2) Record Format: With the consent of the participants we made audio recordings of each interview and then transcribed them verbatim so they could be analysed. It can be difficult to write notes while conducting an interview so the interviewer took only limited notes during the interviews. These were useful for confirming details which would not otherwise have been apparent in the audio recording (eg: body language, context etc.)

Transcribing audio recordings is time consuming. We spent about three hours on transcription for every one hour of recorded audio (plus a significant amount of time coding the resulting text, which was up to 12,000 words long in some cases). To save time we transitioned from verbatim transcription to selective note taking for the final two interviews. We feel this can be justified on the grounds that 1) we had a good understanding of what information was pertinent by this point and that 2) we had reached the data saturation point.

1) Because of the detailed analysis we had already done on the first three interviews we were confident that we could accurately identify the most relevant statements in the interviews, while ignoring less relevant parts of the interviews.

2) The data saturation point occurs when collecting new data ceases to lead to new insights. We felt we began to reach this point after three interviews (see the sample size section for more information). Thus we were confident that the likelihood of major new (unexpected) findings emerging from the final two interviews was low, and therefore that creating verbatim transcriptions for detailed analysis would have produced diminishing returns.

The list below summarises the forms in which we collected data.

- Limited notes taken during interviews
  - Useful for confirming things which were indicated non-verbally by the interviewee (often represented in the transcripts with square-bracketed “editor’s notes”).
- Audio recordings of interviews
  - Provides an exhaustive record of the interviews for later reference.
  - A recording of the final interview could not be made due to technical difficulties at the interviewee’s end.
- Transcriptions of recordings
  - Easier to analyse than the audio recordings. Allows for things like coding and quantitative textual analysis.
  - Transcriptions were not made for the final two interviews (see ‘selective note taking’ below).
  - It is common practice to do multiple passes of the transcription to improve accuracy, but we did not do this as there was only one researcher and we could not spare the time. However, we feel that the risk of mistakes is low as the same researcher did transcriptions as conducted the interview, and the transcription was done within hours or days of the interview (so it was fresh in their mind).
- Selective note taking
  - Used in-lieu of transcripts for the final two interviews only (once we felt we were approaching the data saturation point).
  - Ideas or statements which seemed important (based on our previous and more comprehensive analysis) were noted down.

3) Unit of Analysis: The term unit of analysis refers to the class of discrete entity which is analysed in a study and influences the choice of sampling technique. [28] We considered the following units of analysis for our study:

- Organisations
- Departments
- Roles
- Individuals
- Specific policies

Given the goal of our study we naturally gravitated towards ‘specific policies’, and assumed that we would be able to obtain formal records of network policies from which to extract common themes for analysis. However, initial investigations made it clear that few enterprises document network policy, instead treating the network configuration itself as the only authoritative record of policy. Interpreting network configurations would have been prohibitively labour intensive so instead we chose ‘individuals’. They are able to describe policy examples and discuss their implementation and impact through personal experience. This reduces the cost of labour for us, and gives us access to additional insights, but introduces some additional threats to validity (see below).

4) Target Population: When conducting survey research (of which interview research is a subset) it is important to select a well-defined target population. [28] We determined that network administrators would be the most valuable participants for this study for the following reasons:

- Individuals in this population frequently work at the “code level”, manually configuring network hardware and writing scripts.
- Individuals are responsible for the day-to-day management of enterprise networks.
- More senior workers (ie: senior network administrators) take a longer-term view of network policy and can thus provide context for policy-related decisions.

5) Sampling Technique: Our target population is narrow and well-defined, making purposive sampling favourable. This is “a process where participants are selected because they meet criteria that have been predetermined by the researcher as relevant to addressing the research question (e.g., people of
Given our unit of analysis and data collection technique, sampling our target population required recruiting participants into our study (as opposed to, for example, obtaining a data dump from a database). This raised the difficulty and cost of accessing each data point in our population sample, and thus pushed us towards convenience sampling (“in which research participants are selected based on their ease of availability.”) [30, p. 124]. Aside from personal contacts we did not have a straight-forward way of accessing the target population. Thus, to increase the size of the sample, we also used snowball sampling (where “a small pool of initial informants nominate other participants who meet the eligibility criteria for a study”) [30, p. 815].

In total we asked 15 people to participate in our study. Of these two declined, five accepted, and the remaining eight either stopped replying after an initial indication of interest, or never replied at all. We focussed on organisations located in Australia and New Zealand, and approached organisations we had some link to (e.g. professional contacts with our industry partner, through academic or personal contacts).

In practice snowball sampling was not effective as study participants reported working on very small teams (see the findings section for more information), so they were not able to recommend many additional people for us to interview.

6) **Eligibility Criteria:**

- **Access:** We cannot travel great distances to perform the interview. Participants must either be located near us, or be available for video calling.
- **Language:** Participants must be fluent in English (as we have neither the ability nor the resources to translate responses).
- **Background:** Participants should have at least 1 year of professional experience managing enterprise networks.
- **Experience:** Participants should have been involved in managing the network at their current place of work for at least 1 year (although this need not have been their main job, as sometimes experienced individuals consult/help out with tasks that lie outside their job description).

7) **Sample Size:** Determining a suitable sample size can be complicated. Rather than selecting an arbitrary number, we aimed to reach the ‘data saturation’ point. This is the point at which “researchers sense they have seen or heard something so repeatedly that they can anticipate it”. [30, p. 875] From this point on collecting additional data produces diminishing returns.

Data saturation should not be confused with theoretical saturation, which occurs when additional analysis of the collected data results in few (or no) new insights. Data saturation is a redundancy of information. Theoretical information is a redundancy of analysis. [30, p. 875]

In our case, data saturation occurred after just three interviews. In the fourth and fifth interviews we found ourselves able to anticipate participants’ responses and when we analysed the data from these interviews we found ourselves simply recording additional or corroborating evidence for existing theories. Additionally, no new codes were created based on the data from the last two interviews.

**B. Data Analysis Procedure**

In this subsection we outline the steps we followed to analyse the data we collected. It is important to note that we did not collect all the data and then perform this procedure in a single discrete step. Rather we analysed the data we collected continuously, in a manner more consistent with Creswell’s “data analysis spiral”. [31, p. 150] This allowed us to grow our understanding of policy space (the domain under investigation) gradually and organically, and to feed this growing understanding back into our data collection process (eg: by being able to ask more astute follow-up questions in interviews, by knowing which lines of inquiry to focus on, etc.) See the findings section for the results (outputs) of this process.

1. Inputs: Collected data, i.e. Interview recording, transcript, and/or notes (see the record format section above).
2. Identify significant statements (text segments) and code them [32, p. 14]
   - Text segments were typically a few words long, but in a few cases were 2-3 sentences long.
   - New codes were created whenever several statements sharing a common theme were identified.
   - Codes were constantly refined, and definitions for each were created and updated as more matching statements were identified.
   - Through this refinement some codes began to overlap and were combined. We attempted to strike a balance between a simpler (and hence more usable and reliable) coding scheme, and remaining sensitive to nuance in the data, as discussed in [33] and [34].
   - Some very frequently occurring ‘primary’ codes were specialised into ‘secondary’ codes, creating ‘code families’ as discussed in [34]. For example, the primary code ‘policy driver’ had the secondary code ‘security’ created underneath it.
   - Relationships between codes were established, creating a network. For example, ‘policy driver’ causes ‘policy dynamism’. This network made it easier to understand the differences between codes and apply them reliably. See the appendices for a diagrammatic representation of the code network we created.
   - Over time the codes (and the network linking them) stabilised and fewer changes were made to it.
   - We used the qualitative data analysis computer program Atlas.ti to assist with this process.
• See our discussion on reliability below for more information about our coding process.

3. Policy identification
• Many of the questions we asked participants were designed to (directly or indirectly) elicit descriptions of real-world network policies.
• These policy examples can be used to propose new policy space dimensions (see below).

4. Context mapping
• We asked participants a few questions about themselves and the enterprise they work for. We used this information to better understand the context in which their answers to our other questions were given.
• We recorded information of this nature, allowing us to compare and contrast the people we were interviewing, the enterprises they worked for, and the networks they worked with.

5. Theory generation and proposal of policy dimensions
• As we refined the emergent codes our research questions (see below) codes and questions became congruent. Thus codes became linked to research questions.
• We make some statements or generalisations about the codes thus linked to each research question.
• Thus we generate theories about the policy space domain (e.g. By noticing patterns in network admins’ behaviour or environments), and also propose new policy space dimensions (e.g. By identifying commonalities among policy examples).

6. Follow-up.
• All participants indicated that they were happy to be contacted with follow-up questions.
• We contacted four participants with such questions, to better understand some responses, and to investigate lines of enquiry which we might have missed during the interview.

7. Carefully refine the study based on the analysis we carried out.
• We carefully refined our research questions as we learned more about the domain. Our goal here was to focus the questions without changing their nature or intent.
• We also modified our analysis plan, eg: by choosing lines of enquiry to focus on.
• We look for corroborating evidence across multiple interviews for the theories we generated and the dimensions propose.

C. Pilot Study

After completing the design of our study we conducted a pilot study. This involved carrying out two interviews exactly as we intended to carry out the ‘real’ interviews, including analysis of the data as described above. We used this experience to refine the interview and data analysis process.

This included adjusting the wording and order of some questions to achieve a more natural flow and remove ambiguity for the participants. We did not make any significant changes to the intention or nature of the questions we asked, and in practice the execution of the interviews before and after the pilots was quite similar. The main difference was that the interviews proceeded more efficiently, with less time spent explaining questions and more time spent listening to the participants.

The process of analysing the data from the pilot interviews allowed us to develop our codebook, so that by the time we began the ‘real’ interviews we already had a stable collection of codes. Thus, the great majority of our codes emerged during the pilot process. We did not expect this, however we also reached the data saturation point sooner than expected, which suggests a high degree of agreement among participants.

Initially we planned to use the data from the pilots to refine the study only, and to exclude it from the final analysis. However, the response rate to our study was low (see the sampling technique section above) and the data from the pilots provided a number of interesting insights, so we post-hoc merged our analysis of the pilot and ‘real’ interviews. We believe this choice is justifiable on the grounds that the execution of the interviews did not change substantially after the pilot interviews.

As discussed above, we improved the wording and order of some questions, but this served mainly to improve the efficiency of the interviews. For example, during the pilot interviews we were forced to make ad-hoc changes to some questions as a result of unexpected responses to previous questions, or to make things clearer for participants. We found the process of merging our analysis fairly straightforward, strengthening our belief in this justification.

D. Ethical Issues

This study was reviewed and approved by the University of Canterbury’s Human Ethics Committee (UC HEC) under reference HEC 2017/13/LR-PS. Below is a list of potential ethical issues we identified and information about how we mitigated them.

• Informed consent
  – Participants were given copies of the study information sheet and consent form (see the relevant appendices).

• Audio recording of participants
  – All participants were asked for their consent to be recorded on the study consent form, and then again in person before the interview began.
  – The study information sheet noted that an audio recording of the interview would be taken if the participant consented.
  – Participants were advised before the recording device was activated, and again when it was halted.
  – The recording device was not disguised in any way, and was placed in clear view.
• Disclosure of sensitive information
  – Participants may have been privy to sensitive and/or confidential information at their enterprise (e.g. information about security, client data, business strategy etc.)
  – It was possible that participants might (accidentally or intentionally) disclose information that they or their enterprise considered sensitive.
  – The questions we asked participants were designed to avoid such disclosures.
  – Participants were informed that they were free to withdraw from the study at any time, and that if they did so the data we had collected would be destroyed.
• Participants’ privacy
  – Some participants might prefer not to have their name (or information which could identify them) published publicly. One participant’s enterprise specifically requested not to be identified.
  – No participant or enterprise will be named in any material we make available to individuals outside the research team.
  – We will do our best to limit the amount of identifying information made available to people outside the research team. However, it is impossible to remove all such information. Only information which is relevant to the research will be made available (e.g. enterprise size, complexity of network, enterprise domain etc.)
• Information security
  – The data we collected was stored on secure servers at our institution (the University of Canterbury) and on the principal researcher’s (physically secured) computers only.

E. Validity

There are many views on validation and reliability in qualitative research. [28], [30, p. 714], [31, p. 202]

Easterbrook [28] and Svensson [35] address validity in qualitative research with traditionally quantitative terms such as construct, internal, and external validity. Creswell [31, p. 202] summarises a number of perspectives to validation, recommends that researchers use accepted strategies with which they are comfortable, and presents eight general purposes validation strategies. In [30, p. 714] the authors suggest that qualitative analogues for quantitative methods of validation (like those listed above) have value, but only up to a point. In general they seem to advocate for documenting.

We have adopted views similar to those expressed in [31, p. 202] and [30, p. 714] and agree that it is more important to clearly document our findings and the processes that led to them than to adopt an overly systematic and generalised approach to validation. In other words, we aim to present our process and findings with sufficient detail for readers to make up their own minds. From this it follows that we are aiming for theoretical rather than statistical generalisation. Below we list those of Creswell’s validation strategies which we adopted, and identify a number of specific threats to the validity of our study.

1) Validation Strategies:

• Peer review. We asked two colleagues who were not involved with the study to evaluate its design and provide feedback. Their expertise and background is as follows:
  1. Very experienced researcher with a great deal of experience in conducting qualitative and quantitative research in the field of human-computer interaction.
  2. Post-graduate student with experience conducting survey research.

• Negative case analysis. “The researcher revises initial hypotheses until all cases fit … eliminating all outliers and exceptions”.
  – As we discovered new examples of policies (‘cases’) we ensured that they could be described in terms of the policy space dimensions we had proposed (‘hypotheses’).
  – We introduced new dimensions and refined existing ones until all example policies could be described (see the findings section for specific examples).

• Rich description.
  – In the findings section we provide a description of the study participants, the organisations they work for, and the networks they work with.
  – We hope this will allow readers to evaluate the transferability of our findings to other contexts.

2) Threats to Validity:

• There is the potential for misunderstanding when gathering policies from verbal description (as opposed to formal documentation).
  – Different participants may understand the same policies differently.
  – We may misunderstand what they are telling us.
  – If we were sampling policies from a formally documented repository we could control for sampling bias. But participants will pick only the policies which happen to occur to them, and this will be influenced by the questions we ask in unpredictable ways.
  – But the advantage of an interview is that the interviewer can clarify questions for participants.

• Survey research relies on participants perception of reality.
  – They may not have a realistic, accurate or representative understanding.
  – This would be averaged out over a large sample, but our sample size was small.

• Reliance on semi-structured interviewing as the sole data collection method. [30, p. 423]

• Participant recruitment. To achieve representational validity a survey must have an appropriately large sample
set, and we found that the majority of people approached were not interested in participating.

- Sampling bias. [28]
  - We used convenience sampling, so our sample may not be representative. This means we cannot make statistical inferences to target population [29].
  - To address this we have made it clear what subset of the population is described in the dataset in the section on our findings. Support for this approach can be found in [29], [30, pp. 60–61].
  - Our chosen data collection technique (semi-structured interviews) allows us to make the most of the data points we do have, as in most cases we were able to meet face-to-face with participants, pursue new lines of enquiry ad-hoc, and ask detailed follow-up questions after preliminary analysis. We also feel that, having agreed to take part, participants were more likely to fully engage in the process and give high-quality responses than they might have if we had used a technique like a questionnaire.
  - Despite this issue we believe the sampling techniques we have used are acceptable given the goal of this study as an exploratory investigation.

- Ambiguous questions. While we aimed to develop questions which were unambiguous there is still a risk that not all participants interpreted them all in the same way.

- Confounding variables
  - Perhaps the size of the enterprise network will affect the responses. For example, perhaps small enterprise networks (eg: serving tens of users) differ significantly in how they are managed from large ones (eg: serving thousands of users)

3) Reliability: This refers to how reproducible a study’s results are. [28] Discussions of reliability in qualitative research often focus on inter-coder reliability [31, p. 209] (the extent to which different researchers code the same material in the same way). In our study there was only one coder, so this was not an issue.

Another factor in reliability is code stability (concerned with ensuring that a coder’s use of codes does not change over time). [31], [34] We achieved code stability by writing concise definitions for each code (forming a codebook), by constantly referring to previously coded statements (to make sure our understanding of the code had not changed), and by putting codes in context with one another (see notes about our ‘code network’ above).

IV. FINDINGS

A. Context

Here we present a description of the context in which network administrators create network policy. This is intended to give readers a better understanding of the environment in which network administrators work. We also refer to this context as the “experience” of creating and maintaining network policy.

<table>
<thead>
<tr>
<th>Participant</th>
<th>Number of forwarding devices</th>
<th>Number of hosts</th>
<th>Number of users</th>
<th>Max network throughput</th>
<th>Enterprise domain</th>
<th>Number of staff</th>
<th>Size of network team</th>
<th>Area of network management</th>
<th>Area of enterprise</th>
<th>Years of experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Participant 1</td>
<td>10-100</td>
<td>1000-10,000</td>
<td>1000-10,000</td>
<td>100-100</td>
<td>High school</td>
<td>All</td>
<td>All</td>
<td>All</td>
<td>IT department</td>
<td>&gt;10 years</td>
</tr>
<tr>
<td>Participant 2</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100</td>
<td>Polytechnic</td>
<td>All</td>
<td>All</td>
<td>All</td>
<td>IT facilities departments</td>
<td>10-10 years</td>
</tr>
<tr>
<td>Participant 3</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100</td>
<td>Engineering office</td>
<td>All</td>
<td>All</td>
<td>All</td>
<td>IT facilities departments</td>
<td>10-10 years</td>
</tr>
<tr>
<td>Participant 4</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100</td>
<td>University</td>
<td>All</td>
<td>All</td>
<td>All</td>
<td>IT facilities departments</td>
<td>10-10 years</td>
</tr>
<tr>
<td>Participant 5</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100-1,000</td>
<td>100</td>
<td>University</td>
<td>All</td>
<td>All</td>
<td>All</td>
<td>IT facilities departments</td>
<td>10-10 years</td>
</tr>
</tbody>
</table>

This includes the challenges and pressures faced by network administrators and the processes and technologies they use to address them.

See Figure 1 for a brief summary of the contextual information we gathered from each participant.

The network administrators we spoke to all work in small teams of just one or two, usually (though not always) in conjunction with a larger technical team which is responsible for other IT infrastructure (eg: servers, security). The network administrators thus have wide-ranging responsibilities and work with most or all parts of their enterprise’s network. This wide-ranging responsibility makes delegation desirable. Tasks which might be delegated include the physical installation (as opposed to the digital configuration) of new network hardware, or carrying out work which they understand well and find repetitive (such as configuring ports to provide access to the correct VLANs). However, the people to whom such tasks might be delegated (eg: contractors or less technical colleagues) typically do not have an detailed understanding of the network configuration and often make mistakes, generating more work.

We observed a high degree of agreement among the network administrators we interviewed (that is, few of their statements were contradictory, and most were confirmatory). This suggests to us that the experience of creating and maintaining network policy is consistent across a diverse range of enterprises. Greenberg makes a similar observation in [9]. We note a number of other consistencies:

- All the administrators we spoke to made use of the same relatively small collection of low-level technologies. For example: VLANs, Active Directory, firewalls, access control lists (ACLs), 802.1X, LDAP, and scripting languages like Python and Bash.
- Network policy is informally documented at all of the enterprises in question, with the live network configuration itself being the primary form of policy documentation in most cases.
- The administrators report adopting similar processes for verifying that policies are functioning as intended (mostly
informal manual testing, relying heavily on their detailed knowledge of the network and technical expertise).

- The most frequently modified network configurations were port:VLAN mappings and firewall rules.

Many authors have commented that today’s networks are complex, fragile, and statically configured, making them expensive to maintain and update. [13], [36], [37, p. 74], [6], [9]

This narrative is supported by our conversations with network administrators, which indicate that certain network changes which are important to users are significantly delayed due to the cost of the changes required. The administrators we spoke to identified several reasons for this. Firmware updates are disruptive, being error-prone and usually requiring network downtime; updates for older hardware may be unavailable; the most compelling features of new hardware may not be compatible with old hardware (making it difficult to justify the incremental costs of upgrading the entire network); monolithic upgrades of all network hardware at the same time (sometimes called ‘forklift’ or ‘rip and replace’ upgrades) are effective, but are expensive and necessarily infrequent; it is difficult to coordinate updates to heterogeneous network hardware.

All of these factors discourage frequent network upgrades, and yet our discussions with network administrators showed that network users and enterprises make diverse and frequently changing demands of the network (see our discussion of policy drivers below). We believe that network management today inefficiently employs a static model to address a dynamic problem.

B. Proposed Policy-Space Dimensions

In this section we present the main contribution of this study: A list of orthogonal (non-overlapping) ‘dimensions’ which we believe can, in combination, can represent the vast majority of policies needed to operate an enterprise network.

Each of these high-level dimensions could be implemented in many different ways, in many different networks. We suggest that specific implementations of these dimensions be accompanied by ‘properties’, that is, information specific to the enterprise. Such information could be sourced from the enterprise’s employee databases, official records, historical statistics, static configurations etc. Our proposed dimensions would thus provide a high-level interface for making this information available to the network, enabling more intelligent and dynamic behaviour. We see ‘dimensions’ as invariants, applicable to all enterprise networks, and ‘properties’ as network-specific information associated with a dimension. Below we provide a number of example properties alongside certain proposed dimensions as we introduce them.

In this model, properties of one dimension may be expressed in terms of another dimension (similar to a type system in general purpose programming languages). For example, a network-connected device may be associated with a specific user. Thus, in one implementation the Device dimension might have a ‘user’ property, whose value is expressed in terms of the User dimension. Likewise, the User dimension could have a ‘devices’ property expressed in terms of the Device dimension. This allows dimensions to be used flexibly.

In this report we represent properties with ‘dot’ syntax, where the lower case property name is preceded by the capitalised dimension name (eg: Device.name). When a property is understood to be expressed in terms of a particular dimension it is written inside braces like so: Device.(user:User
\[\rightarrow\], where the lower case property name precedes the capitalised dimension name. Thus, Locus.(source: User
\[\rightarrow\]).role, should be read as “classify the traffic in terms of the Locus dimension, identify the source property of this dimension and express it in terms of the User dimension, and then identify the role property of this dimension”.

We envision these dimensions being used in a high-level network management framework. For example, in a software-defined networking (SDN) context, a program running on the controller could classify packets in terms of these dimensions and apply high level policies (written in the same terms) to them. Policies need not involve all the dimensions we present here, but could be composed only of a relevant subset. Granular policies expressed in terms of these dimensions could be combined into more complex policies using predicates like Equals, Or, And etc. and conditionals like If. Note that we do not specify any particular policy description language (PDL) [17] or format, but rather present these dimensions as the core concepts that such a PDL should be able to express.

Summary:

- Dimensions are essential concepts common to most enterprise network policies.
- It should be possible to express the great majority of enterprise network policies in terms of the dimensions we present here.
- Dimensions are abstract and high-level, and are applicable to most enterprise networks.
- Properties are specific pieces of information within each dimension.
- The properties provided by each dimension are implementation-specific. Thus, different implementations of our model may make different properties available.
- Properties expose more specific (lower level) information to the network management system. This information may be source, for example, from internal databases.
- In this report we use ‘dot’ syntax as a shorthand for referring to properties. Eg: Locus.(source: User).
\[\rightarrow\] role
- A property of one dimension may be expressed in terms of another dimension (eg: User.(device: Device))

1) User: This dimension captures the concept of a network user, allowing network policy to address specific individuals. Example use cases:
Traffic could be tied to known users for purposes such as accountability and security.

The network could query the state of the user to determine when they were last active (this would be useful for policies like ‘after 10 minutes of inactivity sign the user out’).

Example properties:

- Role (eg: student, staff, manager). Roles could be hierarchical, with higher roles being a superset of lower roles as in role-based access control (RBAC). [38]

2) Device: This dimension provides an abstraction for devices connected to the network. Example use cases:

- Tying traffic to an originating device.
- Tying devices to a particular user.

Example properties:

- Device classification (eg: servers, loaner devices, lab computers)
  - An administrator-assigned tag for devices, allowing multiple devices to be addressed at once (similar to roles for users).
- Device operating system
- Time since last active
- Classifications or judgements created by other network services, eg: infected with malware.

3) Locus (or Directionality): Policies often need to express relationships between two or more entities in a network. Many policy description languages have a similar concept [17]. For example, policies written in the Ponder language [21] have a ‘subject’ and a ‘target’, allowing the expression of ‘to-from’ relationships between network entities in which one is an actor and another is being acted upon (this may be as simple as one device sending traffic to another). This is one possible implementation of this dimension, but it is not the only one. Example use cases:

- A policy which prevents a specific user from sending traffic to a specific server, but which allows the same server to send traffic to that user.
- Port access control. Allow only certain kinds of devices to connect to the network via certain ports only.

Example properties:

- Source (could be expressed in terms of the User or Device dimension, or as a collection of Traffic Properties, as described below).
- Destination (as above). In the case of multicast and broadcast transmissions there might be more than one destination.
- Path. The path the traffic was (or will be) forwarded on through the network.
- Info. Additional information relating to the forwarding path (edge switches could populate this with information explaining their decision making process for the benefit of the network management system).

4) Traffic properties: This dimension encapsulates the various low-level attributes of the traffic switched by the network, allowing network policies to refer to information in packet headers.

Example use cases:

- Referencing the transport protocol (WiFi, ethernet) would allow things like preventing a user from accessing the wireless at certain times (eg: during exam conditions at an educational institution).

Example properties:

- Protocol (eg: TCP, UDP, HTTP, IP, MAC, ICMP, BGP, SSH, SFTP, SNMP)
- Bandwidth
  - Throughput over a certain time period
  - Cumulative bandwidth usage over a certain time period
- Direction (in/outbound from/to a device, or the network itself)
- Destination URL (for HTTP traffic)
- Traffic source
  - IP address
  - MAC address
- Traffic destination
  - IP address
  - MAC
- IP address (source or destination)
- MAC address (source or destination)
- VLAN tag
- Flow (allow policies to capture packet flows, rather than just individual packets)

5) Physical location: This dimension ties traffic to a particular location. This could be geographic, or enterprise-specific (eg: block C, room 123).

Example use cases:

- iOS devices present AirPlay devices in a system menu, but if 1000 Apple TVs are installed across campus they would be overwhelmed. If the network was location-aware it could display only the devices close to users (eg: in the same meeting room as them, or on the same floor).

6) Temporal: This dimension allows for the expression of policies which apply only at certain times. Example use cases:

- Apply a policy from a given start date to a given end date.
- Specify a duration for which a policy should apply (eg: “for 5 hours after the main doors are unlocked”).
- Applying a policy for the duration of an event specified by some other network-connected system, such as a shared calendar.
7) Authentication: Network policy frequently deals with authentication. In our model this dimension would allow the construction of policies which are authentication-aware.

Example properties:
- Authenticator (which network service carried out the authentication?)
- Authentication method (RADIUS, WPA2, LDAP etc.)
- Abstract privileges granted by some authority.
  - Eg: Rather than a static policy which says “users X, Y, and Z may access service S” we could create a more general policy which says “before accessing service S authentication.privilege must match description D”.
  - Any device which can produce a signed certificate (or similar) can thus demonstrate that it has been given permission to do XYZ, without needing to identify itself as operating under the auspices of any particular user, or as being any particular device, or even as being any particular type of device.
  - Thus in our model of policy space, ‘authentication’ need not be tied to specific user or device.
  - This could simplify network management in a number of ways. For example, the physics department at a university could issue some students with physical or virtual access tokens to grant them temporary access to a secure lab (without needing to talk to the IT department to have a flag on the students’ accounts updated).
  - This demonstrates the flexibility of our model.
  - In an SDN environment such authenticated privileges could be published in many different ways, so that users do not have to worry about installing certificates or entering pass phrases.

8) Trigger: From our discussions with network administrators we determined that policies should be able to respond to predefined events. These events could be Example use cases:
- Make a policy active for five minutes after a user account is created (note that this example also includes the ‘temporal’ dimension).
- Deactivate a policy after another policy becomes active.
- Respond to security incidents or network faults.

Example properties:
- Trigger priority.
  - Some triggers could be considered urgent and be prioritised for response by the network management framework.
- Issuing system (the system which issued the trigger).
- Payload data (additional information which the network management system can analyse to determine an appropriate response).

9) Action: Many of the policies we identified through the interviews we conducted include notions of an action or a response to some condition (which could be described in terms of the dimensions we present here). We thus propose a corresponding ‘action’ dimension. Different actions could be available in different networks, depending on the implementation and capabilities of each. In the model we propose here, each concrete action offered by a network would be a property of the action dimension. Example use cases:
- Prevent certain devices from sending traffic to one another (this could be used to achieve traffic isolation, without prescribing a certain low-level implementation strategy like VLANs).
- Redirect traffic through network middleboxes to apply additional services (this could support a service-function chaining architecture [39])

Example properties:
- Drop packets
- Allow packets
- Duplicate packet
- Edit packet (this would also allow redirecting a packet)
- Run a script.
- Notify a human.
- Modify the configuration of the network

C. Examples
We identified approximately 40 real-world examples of policies through our discussions with network administrators. Below we provide a description of each of them (in plain English, paraphrasing the descriptions given to us by the network administrators we interviewed) and show how each may be expressed using only the dimensions we proposed above. Note that we are not trying to detail complete policies written in a specific implementation of our model. Rather we are simply giving examples of real-world policies and identifying the dimensions which would be needed to express them using our model.

1. When a user account is created or deleted in the student management system also update the lists of accounts in Active Directory, Office365, and Google (corporate).
   - Trigger: User account creation or deletion in the student management system.
   - Action: Run scripts to update the relevant services via their published APIs.

2. During exam conditions students may not use the school WiFi network (ethernet only), and may only access specific services (eg: Google Docs).
   - User: If user.role is student.
   - Temporal: If event (exam) is in progress.
   - Traffic properties:
     - If Traffic.protocols contains Ethernet and HTTP/S.
     - If Locus.(destination: Traffic).url is www.docs.google.com
1. Only teachers and year 12 and 13 students are permitted to book Chromebooks owned by the school.
2. Students using loaned Chromebooks are signed out after 10 minutes of inactivity, to allow other users to use the device.
3. Students may not access Facebook during class time.
4. When a student visits a website on a hard-coded list (eg: a list of known porn sites) the dean responsible for them gets an email.
5. Students are allowed to access the internet and some internal services only, and may only use 'web' protocols.
6. TradeMe is blocked during school hours for most users, but allowed for some others.
7. mega.co.nz is blocked for most users, but allowed for some.
8. VPNs and proxies are blocked.
9. The Bonjour protocol is blocked.
10. BitTorrent protocol is blocked.
16. Certain VLANs should be accessible only via certain physical ports.
   a. Student devices should only be able to access the student VLAN from ports in student areas (eg: not from the staff room).
      - User: If User.role is student
      - Location: If Location.classification is student
      - Action: Allow packets, else drop packets
   b. Wireless access points provide access to only one VLAN each, and the physical port to which they are connected should only be enabled for the VLAN they need.
      - Device: If Device.mac is 11:11:11:11:11:11
      - Action: Set Locus.path.first_hop.port. ↪ vlan to student wifi.

17. VOIP phones are allowed on any port.
   - Device:
     - If Device.type is VOIP phone.
   - Action: Set Locus.path.first_hop.port. ↪ allowed_ports
   - Locus: If Device.allowed_ports does not contain Locus.path.first_hop.port.
   - Action: Drop packets, else allow packets.

18. a. Only the on-site backup servers are permitted to send traffic to the off-site backup repositories.
   - Device:
     - If Locus.{source: Device}.classification is backup server
     - If Locus.{destination: Device}.classification is backup server and Location is not on-site.
     - Action: Allow packets, else drop packets.
   b. No Windows devices should be permitted to send traffic to the on-site backup (to reduce the chance of malware infection).
      - Device:
        - If Locus.{destination: Device}.classification is backup server
        - If Locus.{source: Device}.operating_system is Windows
        - Action: Drop packets

19. If a student user successfully authenticates to the staff wifi LAN (eg: by guessing the password) redirect them to an ‘access denied’ web page.
   - User: If User.role is student and Traffic: - If Traffic.LAN is staff and Traffic.protocol is HTTP.get - Action: Send an HTTP 302 redirect to the source device.

20. a. Casino management staff have unrestricted access to the internet.
   - User: If User.role is management
   - Action: Allow packets
   b. Some staff have no internet access as their job does not require it (eg: housekeeping, food and beverage, kitchen staff, gaming staff).
      - User if User.role is housekeeping, food and beverage, kitchen, or gaming.
      - Action: Drop packets.

21. Users are classified into three groups and general access policies are applied to these groups (as follows).
   a. Power users have mostly unrestricted access to the network and the internet.
      - User: If User.classification is power user
      - Action: Allow packets.
   b. Standard users have mostly unrestricted access to the internet, but can’t access things like pornography, and can’t download very large files (but throughput is not throttled).
      - User: If User.classification is standard user
      - Action:
        - Redirect traffic to vendor-supplied proxy server for filtering.
      - Traffic: If Traffic.protocol.http.response.file.size is greater than 2GB.
      - Action: Drop packets.
   c. Intranet users can only access the intranet and a small number of business-related sites (eg: ACC).
      - User: If User.classification is intranet user
      - Traffic:
        - If Locus.{destination: Traffic}.ip_address is not in range 172.16.0.0/12
        - Or if Locus.{destination: Traffic}.url is not in [www.acc.co.nz, ...]
      - Action: Drop packets.

For each of the remaining policy examples we simply list the dimensions which would be used to represent them.

22. User-supplied devices may connect to the network, but may only access the internet (and strictly no local devices).
   - Device, Locus, Traffic, Action

23. Casino-owned devices with valid security certificates: Can connect to the private WiFi, get unrestricted access
to the internet, get very limited access to the corporate LAN (email, and rostering website only).

- Device, Authentication, Traffic, Action

24. Casino uses a third party service to identify problem gamblers. A server monitors various signals and sends messages to hand-held devices to alert staff. Firewall rules prevent those devices from sending traffic anywhere other than the identification server.

- Device, Action

25. Normal internet traffic must pass through a proxy server (which enforces a number of other policies) to get to the internet.

- Locus, Traffic, Action

26. User-supplied devices may not send traffic to the IT department’s servers.

- Device, Action

27. Staff on enterprise-owned laptops (with valid security certificates) can connect to their network-mounted ‘H’ drives.

- User, Device, Authentication, Action

28. Students can access the internet and some internal servers.

- User, Device, Locus

29. Physical kiosks are available on campus for users to reset their passwords. These kiosks can only send traffic to a specific server.

- Device, Action

30. Visitors to campus should have (bandwidth-limited) internet access.

- User, Traffic, Action

31. Staff on enterprise-owned laptops (with valid security certificates) can connect to their network-mounted ‘H’ drives.

- User, Device, Authentication, Action

32. It should be possible to send emails from the photo kiosk at the on-campus pharmacy.

- Device, Location, Traffic, Action

33. Users should not be able to set up their own DHCP servers

- Traffic, Action

34. Users should not be able to attach their own downstream switches to the network.

- Traffic, Action

35. Running proxy ARP on certain networks is banned.

- Traffic, Locus, Action

36. Salespeople should not be able to access the network engineering wiki.

- User, Locus, Action

37. Only staff members who have filled out a policy form are allowed access to the internal network. Otherwise staff are only given access to the internet.

- User, Locus, Action

38. Only corporate (not personal) devices may use the corporate network.

- Device, Authentication, Action

In some cases it was difficult to separate the policies network administrators described from the specific implementation they had used. Below we describe these policies, and show how we might redefine them in our model.

1. Scenario:
   - There is a VLAN for a specific building (the event centre).
   - This VLAN is only available to devices inside this building.
   - This VLAN has a route to the internet (and any devices connected to this VLAN can access the internet).
   - Devices can connect to this VLAN via a public wireless access point.
   - The main network firewall normally applies a lot of rules to traffic, but the event centre has its own firewall and this only filters pornographic or racist websites.

- The policies we might implement:
  a. Users should only be able to access the internet if they have logged into the enterprise portal.
     - User: If User.{auth: Authentication}.↩→ status is logged in.
     - Action: Allow packets, else drop packets.
  b. Any device connecting from the event centre may access the internet.
     - Location: If location is event centre.
     - Locus: If Locus.{destination: Traffic Properties}.destination_IP is not an IP address on the local network.
     - Action: Allow packets, else drop packets.
  c. Traffic from anywhere other than the event centre should be filtered through a middlebox.
     - Location: If Location.building is not event centre.
     - Action: Redirect traffic through the desired middlebox.

We also discussed policies that network administrators would like to implement but for some reason (eg: technical limitations, budget constraints) are currently unable to do so.

1. There is a VLAN for staff users and a VLAN for student users. Users should only use the correct VLAN for their role.

- The current implementation is to have separate SSIDs for each class of user, with WAPs for each which are connected to a physical port configured to forward traffic to the appropriate VLAN. Users are only given the password for the VLAN appropriate for their role.
  - This policy is fragile (potential for misconfiguration of port:VLAN mappings) and inefficient (more WAPs needed).
• The network administrator we spoke to would prefer an implementation which determines the appropriate VLAN for a user based on their login credentials and which arranges for their traffic to be forwarded to this VLAN only.
  – This could be expressed in our model using the following dimensions: User, Traffic, Action

2. Some network-connected devices (eg: security cameras, routers, switches, sensors) expose management interfaces to the network, often implemented as an HTML page served by the device. These pages may allow users to view configuration details, network information, or even reconfigure the device.
  • Currently access to these interfaces is blocked by manually reconfiguring the devices which expose them, but as this is a manual process sometimes some devices are missed.
  • It is difficult for the network to enforce this kind of policy as it requires host-side changes to implement. However, one approach might be to scan all network attached devices for signs of exposed management interfaces, or to prevent HTML traffic being sent to certain classes of devices.
  • Another approach might be to have the network look for traffic which indicates the presence of such interfaces. So as soon as a user accesses such an interface the network sends an alert to an administrator, or blocks the traffic itself. This policy could be expressed using the following dimensions: User, Device, Traffic, Action

V. APPENDIX 1: INTERVIEW QUESTIONS

1. a) What kinds of users do you have on your network?
   b) What kinds of devices do you have on your network?
   c) What sorts of restrictions are applied to users and devices on your network? What are the things which they may and may not do?

2) Are those restrictions applied uniformly, or only under certain conditions? Suggest examples from the list below if necessary.
   • Based on time of day
   • Based on the user’s role (manager, engineering, network admin)
   • User’s department
   • User’s assigned project(s)
   • Payment/billing/account data
   • User location (campus, building etc.)
   • Method of connection to the network (direct Ethernet, remote VPN etc.)
   • Type of user device (company supplied, or user supplied?)

3. Could you describe two or three specific policies which have been implemented in your network?

4. Can you describe an event (or several) which resulted in new policies being created or existing ones changed? Use the following as conversation starters to fill in any gaps (especially if the interviewee is reticent). Ask interviewee to discuss example scenarios if necessary.
   • Were there ever legal reasons for any of the network policies you implemented? Eg:
     – Privacy
     – Auditing (enterprises are often required to make data accessible to auditors)
     – Safety
   • Administrative changes
     – Departments moving floors or buildings
     – Users taking on new roles (eg: becoming a manager)
     – Directives aimed at improving efficiency
   • Do you implement policies in order to provide employees with resources necessary for them to do their jobs?
   • Do you implement policies to make some other stakeholder happy? Eg:
     – Make customers happy (eg: internet access while on-site)
     – Provide visiting board members with access to network resources
   • Do you implement policies in response to network incidents/faults? Eg: Phishing websites, data exfiltration, misuse of the network.

5. a) How often are policies created or modified? Per day/week/month/year.
    b) Are some kinds of policies created or modified more often than others? If so, why?

6. a) Does creating or modifying policies create a lot of work for you? Is this a major part of your job?
    b) How much work does it typically take to create or modify a policy?
    c) Can you tell me about a time when creating or modifying a policy was particularly expensive? This could include initial or ongoing costs, and could be in terms of man-hours or resources like CPU time.

7. How did you implement the policies we’ve discussed? What processes and tools did you use? What was your workflow?

8. What tools and technologies actually enforce the policies you implement? Firewall rules, ACLs, honour system, topology reconfiguration.

9. a) How are policies recorded? Is there a format? This may come up when discussing previous questions. Suggest examples from list below if necessary.
   • Formally defined and recorded (ie: documented in a consistent format such as a policy description language, or a formal specification document)
• Informally defined and recorded (eg: documented in prose, or some format which can be stored and later interpreted, but which is not necessarily consistent across all policies)
• Informally defined and not recorded (eg: policy is known to network administrators and noted mentally)
• Not specified in any way/question not applicable because there are no policies/restrictions

b) Could you provide an example?

10. How do you know if policies are implemented successfully or not? Is there a process for verifying them?

Suggest examples from the list below, if necessary.

- With a formal and automated testing framework which runs against the network.
  - If yes, specify if it runs against a production or a test network.
- By hand (network admins attempt actions manually and see if they are allowed/disallowed as expected)
- By sight (network admins review the network configuration and determine if it looks like it should operate as expected)
- Not at all
- A mixture of the above options (ie: some policies are tested one way, others another; please specify)

11. Are there policies you would implement if you had the right tools?

VI. APPENDIX 2: INTERVIEW PROCEDURE

1. Before the interview the participant was sent copies of the study information sheet and consent form. These:
   - Explained the purpose of the study.
   - Identified the researchers.
   - Explained that participation was voluntary and that the participant could withdraw at any time without penalty, and that any information they had provided would be destroyed.
   - Stated the duration of the interview (40-60 minutes, plus up to 15 minutes of follow-up questions via email).
   - Stated that the participant could request to view the results of the study.

2. At the beginning of the interview the goal of the study was again summarised for the participant and the participant was asked to verbally confirm that they had read the information sheet and that they were happy to proceed with the interview. The interviewer confirmed that the participant had signed the consent form.

3. The interviewer conducted a brief sound check with the recording equipment.

4. The interview was conducted.

5. After the interview the interviewer thanked the participant and asked:

VI. APPENDIX 3: CODEBOOK

- Ad-hoc interview question
  - An interview question the interviewer invented during the interview (as opposed to one which was written before the interview began). Asked to pursue an interesting line of enquiry which differs from the spirit of any of the prewritten questions. Thus, this excludes questions asking for clarification and follow-up questions which have the same theme/spirit as the original (prewritten) question.
- area of work (within network management)
- availability of resources
- bandwidth
- challenges in network management
  - Situations which are generally challenging in network management. Likely to be a challenge in traditional networks and SDN.
- class of host
  - Different classes of host devices (eg: phones, laptops, cameras, servers, PCs.)
- class of policy
- class of user
- cost of policy
  - The costs associated with a policy (implementation, maintenance etc.)
- enterprise requirements
  - Things that users don’t have a direct interest in, but which the enterprise must provide. Eg: Access to the network for air conditioning system.
- firewall rule
- frequency of policy modification
- general approach to network management
- honour system
  - Network users are asked not to do something.
- implementation strategy
  - The mechanisms by which policies are enacted/enforced in the network.
- Information about the enterprise
- information about the interviewee

6. The participants were sent a final email which:
   - Thanked them again for their participation.
   - Reminded them that they could request to view the data that had been collected from them.
   - Remind them that they were entitled to view the results of the study.
information about the network
internal threat
  – When users deliberately use the network for purposes the enterprise explicitly disallows. This is not limited to security issues, but might also refer to users accessing social media.
interview question
  – A predefined question written before the interview began, and as a part of all interviews.
keeping the network operating smoothly
  – Network admins sometimes need to restrict user behaviour because it can interfere with the operation of the network.
legal requirement
manual inspection
manual testing
manual white/blacklisting
  – Eg: MAC auth, manually updated access control lists
network incident
  – Something happened which prompted the enterprise to institute some policy. Eg: A user was misusing the network.
network monitoring and manual intervention
  – Automated or manual monitoring of the network to make human admins aware of anomalous events so that they can take action.
network protocol
  – A network protocol used in the network. Useful to keep track of these for our reference.
network service
  – An example of a service being run on or provided by the network.
number of forwarding devices
number of hosts
number of network admins
number of network users
Number of staff
packet inspection
parts of the network the admin works with
people involved
  – Who is involved in and/or consulted during the implementation process. Eg: Management, users, network engineers,
policy condition
  – Condition for the application of a policy. Eg: Time of day.
policy driver
  – Policy drivers aren’t policies, although they result in the creation of policies. They may be objectives which policies can achieve, or demands or requirements which policies can satisfy. Eg: “We want to keep things simple for network users, so they’re not tripping over complex environment configurations.”
– Staff rostering system (hosted on a web server)
– Other departments making demands of the network (eg: facilities department at EIT wanting newer IP-based and wireless technology which requires network access)
– PDX (phone system) old (1990s) needed replacing. Brought in Skype for Business (also partly a request from someone important in the organisation who was sick of writing emails and wanted to leave voice mail instead).
– Need for building management devices (network attached)
– Accounting (eg: costs should be charged back to departments, users etc.)
– Accountability (eg: users should be accountable for misuse)
– Enterprise administration (eg: department moves building, floor, or adds a second building/floor)
– Public image, eg:
  – Looks bad to have enterprise IPs in torrent swarms.
  – Looks unprofessional to have exposed web interfaces for network devices (eg: cameras, routers). Same for SNMP streams.
– Network scaling, eg:
  – Adding ports for more hosts once a switch is fully allocated.
  – Adding IP addresses once a subnet is fully subscribed.

B. Network incidents

– Teachers playing movies for students because they had a TradeMe auction finishing).
– One user gained access to a local management account on a lab machine and used this to snoop on the network.
– Student brought a worm in. Led EIT to create ACLs to restrict traffic flow between staff and student VLANs.
– Misuse of the network (eg: accessing inappropriate websites)

C. Trust

– Interviewees said users were trusted to an extent, meaning that they did not feel it was necessary to adopt lots of restrictive policies.

D. User requirements/requests

– Some users make legitimate use of mega.co.nz (fast way to share large files within New Zealand)
– Users want to be able to plug IP phones into any physical port.
– Facebook was initially blocked at EIT, but staff requested it “so they could liaise with students”.
– BYOD
  – Creating hosts that need to be accessed from the internet
  – New projects (lots of work at the start of term for network admins)
  – Company executives visiting office and wanting special access/exemptions.
  – Desire for wireless access (over wired)
  – Users running on-site events (eg: clubs day, conference)
  – Users want access to all the resources their various roles require at all times, not just when they’re in a particular building (mechanical engineering, biology).
  – Users don’t want to be obstructed by network policies. Network policy should not be noticeable to users in the course of their daily work, where possible (ie: they shouldn’t have to think about working around network restrictions to get their job done)

E. Keeping the network operating smoothly

– Enterprise wants to keep devices on the VLANs to which they are allocated if possible, as users can often plug into the wrong port and if that port is enabled for the wrong VLAN for that user then they will generate some unnecessary traffic on that VLAN (eg: broadcast), and when lots of users do this this increases strain on the network and adds to management overhead (lots of unrelated packets and logs on the network).
– A need/desire for backups in case of data loss (eg: due to user error or natural disaster).
– At EIT some printing software gets confused unless all printers are on same /24 subnet.
– To restrict outages to sub-sections of the network (rather than the whole thing) EIT puts blocks/buildings on their own VLANs (as users frequently create loops by plugging the wrong things into the wrong places).
– Availability of resources, eg:
  – Can’t implement some policies with only a small team
– Wireless signal strength/coverage issues
– User activity impacting the network
  – Chatty protocols like Bonjour on Apple devices
2.4GHz video cameras interfering with the wifi spectrum
- Downstream DHCP servers
  - Users attaching wifi routers via a LAN port instead of WAN (router tries to do DHCP over corp network)
- Developers testing DHCP servers
- Developers running proxy ARP (answers on ARP requests on behalf of another device)
- Congestion/QoS was not a driver.

F. Third party requirements
- If have call quality issues third party vendor will ask ‘are you running QoS?’ and if not won’t take seriously.

G. Security
- Students often obtain the password for the staff WiFi LAN (by cracking WPA2), so there are additional mechanisms for preventing them from abusing this access, should they obtain it.
- Information can leak (eg: consider publicised case of ACC leaking data)

H. Negative Examples
Below are factors which we thought might be drivers, but which participants indicated were not.
- Safety (interviewees said not handled by the network, does not influence network policy)
- Privacy (interviewees said not handled by the network, does not influence network policy)
- Legal (according to interviewees legal wasn’t a driver)
  - Copyright legislation (although in interviews admin said legal wasn’t an issue as they classify themselves as an ISP and delegate all responsibility to end users)
  - Auditing (interviewees said not handled by the network, does not influence network policy)

X. APPENDIX 6: LIST OF TECHNOLOGIES CURRENTLY USED TO IMPLEMENT POLICIES
Specific technologies used to enforce implemented policies.
- Scripts
  - Python
  - Batch
- Active Directory
- Linewize
- Packet inspection
  - Some enterprises avoid this due to privacy concerns.
- Local proxy server
- Proprietary products
  - Cisco Meraki (mobile device management tool)
  - Novell Zenworks (desktop management tool: tracks software deployment, handles software licensing)
  - Cisco AnyConnect
  - Web Marshall (web proxy)
- Proxy
- VPN
- VLANs
- DMZ
- ACLs
- Content Keeper
- Firewall rules
  - Including content filters (for malicious or inappropriate websites) from the firewall vendor.
- Honour system
- Manual white/blacklisting
- DHCP snooping
- Topology changes
  - Link capacity limitations
  - Network partitioning (subnets, VLANs)
- Authentication
  - LDAP
  - 802.1X
  - MAC
- Security group tagging
- Cisco ICE

Analysis:
- A lot of the identified technologies are pretty low level.
- Proprietary solutions are often more expressive than alternatives and include higher level abstractions, but limit admins to a single vendor. Admins who prefer (or must use) multi-vendor networks end up thinking only in terms of the low level concepts supported by their multi-vendor environments. Thus, ‘expensive’ dimensions like ‘location’ may be underrepresented in policy examples.
- One participant works with higher level tools than some other enterprises, possibly because they have embraced a single network hardware vendor.
- One participant talked about some new switches which have a web interface.
  - But he doesn’t necessarily like this, as he likes to clone switch configurations.
  - Has to have a TFTP server, export config, manually edit hostname and IP and upload to switch.

XI. APPENDIX 7: EXAMPLE POLICY IMPLEMENTATION WORKFLOWS
Task: Prevent network users (staff and students at Burnside) from accessing ‘time wasting’ websites.
Process:
- Once every two weeks (if admin has time) review internet logs.
- Inspect the top-20 sites (by bandwidth) and determine if any are ‘time wasters’.
– Block such URLs.

• Inputs:
  – List of top-20 most-used sites (by bandwidth) for a recent time period (eg: two weeks)

• Outputs:
  – Regular network users cannot access ‘time waster’ websites.

• Task: Make sure network policy cannot be circumvented with the Tor browser.

• Process:
  – Manually run Tor browser once a week and see if it can access the internet.
  – If it can, tell the network management middleware vendor.
  – Vendor inspects logs and blocks Tor.

• Outputs:
  – The Tor browser cannot be used to access the internet.

• Task: Generic (high level process for any task)

• Process:
  – Discuss with boss (another network admin)
  – If policy may impact network users (eg: a change to Burnside’s student management system): Meet with affected users’ department and discuss.
  – If impact of policy on network users will be minimal then proceed without additional discussion.
  – Implement the policy using the appropriate technology (eg: firewall rule).
  – Add a description of the change to a changelist (spreadsheet).

• Inputs:
  – Problem description
  – Feedback from impacted users

• Outputs:
  – Policy has been implemented
  – Changelog entry

• Task: Add a basic firewall rule

• Process:
  – Open ASDN (graphical firewall interface)
  – Make change (2 minutes)

• Inputs:
  – Requirements for rule

• Outputs:
  – New firewall rule has been implemented

• Task: Apply a consistent QoS policy for voice calls (for Skype for Business at EIT)

• Process:
  – Work out commands for 4 different operating systems on switch hardware (documentation no longer available for some of them)
  – Apply QoS queue to egress. . . Work out how to have a consistent QoS policy across all switches.
  – Voice > Video > File transfer priority

• Inputs:
  – DSCP: Application layer sets a flag based on Active Directory GPO.

• Outputs:
  – QoS prioritisation

• Task: Troubleshoot a connectivity issue for a user

• Process:
  – Review job from user
  – Sniff traffic (Wireshark)
  – Determine if traffic is going where it should be going
  – Reply to user to tell them if the issue is with the network or with the server they are trying to connect to.

• Inputs:
  – Issue description from user
  – Traffic analysis

• Outputs:
  – A determination of the source of the fault

• Task: Reconfiguring the firewall

• Process:
  – Run a shell process which pings an external source at regular intervals and which emits an alert noise whenever a ping is unsuccessful
  – Prepare the new or modified firewall rules.
  – Manually inspect the changes.
  – Review output from a GUI tool which identifies conflicting rules.
  – Make the changes.
  – As the new rules are pushed out, listen for alert sounds.
  – Immediately (manually) roll back the changes if too many alert sounds are observed.

• Inputs:
  – Ping tool
  – Highlighted conflicting rules

• Outputs:
  – Firewall rules

• Task: Implementing a new policy

• Process:
  – Team receives instructions from management about desired policy.
  – Team discusses implementation options.
  – An appropriate number of people are assigned to implement the chosen option.

• Inputs:
  – Policy brief from management (no formal semantics or format).

• Outputs:
  – Arbitrary modifications to the network.

Analysis:
• Policy implementation workflows are ad-hoc and lack structure. They do not have clearly defined inputs and outputs.
  – Contrast with the Agile methodology from software engineering:
    * Inputs:
      · Sprint goal.
      · User stories which deliver value.
      · Acceptance criteria.
      · Additional information from product owner (note, the PO is formally defined/appointed).
    * Outputs:
      · Source code which satisfies the ACs.
      · Automated tests which ensure that the code behaves as prescribed by the ACs, does not introduce bugs in existing functionality, etc.
  – Hypothesis: These unstructured workflows are a symptom of the generally unstructured ways network admins think about network policy. Formal definitions of policy dimensions may lead to more structured and efficient workflows.

XII. APPENDIX 8: STUDY INFORMATION SHEET
See Figure 3.

XIII. APPENDIX 9: STUDY CONSENT FORM
See Figure 4.

XIV. APPENDIX 10: TIMELINE

- Study design begin
- Study design complete
- 2017-03-29: Pilot interview 1
- 2017-03-30: Pilot interview 2
- Study refinement
- 2017-05-02: Interview 1
- 2017-05-03: Interview 2
- Data saturation point reached
- 2017-05-17: Interview 3
- 2017-05-30: Analysis complete

REFERENCES


Andrew Curtis-Black

An Enterprise Policy Description Framework for Software Defined Networking
Information Sheet for Interview Participants

Andrew Curtis-Black is a PhD student at the University of Canterbury working under his supervisors, Prof. Andreas Willig and Dr. Matthias Galster. His research focuses on the applications of software defined networking (SDN) to enterprise network management. This interview study aims to identify essential concepts used to express network policies in industry.

If you choose to take part in this study, your involvement in this project will be to take part in a structured interview lasting no more than 60 minutes. The interview will take written notes throughout the interview and, with your consent, will take an audio recording of the interview which he will later transcribe. With your consent, you may be asked to answer follow-up questions via email. This will take at most 15 minutes of your time. You will only be contacted if you specifically opt-in as the consent form.

In the performance of the tasks and application of the procedures there are no risks to yourself or your institution or organization.

Participation is voluntary and you have the right to withdraw at any stage without penalty. You may ask for your raw data to be returned to you or destroyed at any point. If you withdraw, I will remove information relating to you. However, once analysis of raw data starts (estimated for 3rd May 2017), it will become increasingly difficult to remove the influence of your data on results.

The results of the project may be published, but you may be assured of the complete confidentiality of data gathered in this investigation. Your identity will not be made public without your prior consent. To ensure anonymity and confidentiality, all data will be stored securely on servers owned and operated by the University of Canterbury which are physically located on-campus. Only the researchers (Andrew Curtis-Black, Prof. Andreas Willig and Dr. Matthias Galster) will have access to your data, in addition to the administrators of the university’s servers (who are trusted employees of the university). All data will be destroyed after a period of ten years. A thesis is a public document and will be available through the UC Library.

Please indicate to the researcher on the consent form if you would like to receive a copy of the summary of results of the project.

The project is being carried out as part of a PhD project by Andrew Curtis-Black under the supervision of Prof. Andreas Willig and Dr. Matthias Galster, who can be contacted at andreas.willig@canterbury.ac.nz and matthias.galster@canterbury.ac.nz, respectively. They will be pleased to discuss any concerns you may have about participation in this project.

This project has been reviewed and approved by the University of Canterbury Human Ethics Committee, and participants should address any complaints to The Chair, Human Ethics Committee, University of Canterbury, Private Bag 4800, Christchurch (human-ethics@canterbury.ac.nz).

If you agree to participate in the study, you are asked to complete the consent form and return it to the interviewer.

Fig. 3. Information Sheet

An Enterprise Policy Description Framework for Software Defined Networking

Consent Form for Interview Participants

☐ I have been given a full explanation of this project and have had the opportunity to ask questions.
☐ I understand what is required of me if I agree to take part in the research.
☐ I understand that participation is voluntary and I may withdraw at any time without penalty. Withdrawal of participation will also include the withdrawal of any information I have provided up to the point of withdrawal should this remain practically achievable.
☐ I understand that any information or opinions I provide will be kept confidential to the researchers and the administrators of the University of Canterbury’s servers and that any published or reported results will not identify the participants or their institution. I understand that a thesis is a public document and will be available through the UC Library.
☐ I understand that all data collected for the study will be kept in locked and secure facilities and/or password protected electronic form and will be destroyed after ten years.
☐ I understand the risks associated with taking part and how they will be managed.
☐ I understand that I can contact the researcher Andrew Curtis-Black or supervisors Prof. Andreas Willig and Dr. Matthias Galster for further information. If I have any complaints, I can contact the Chair of the University of Canterbury Human Ethics Committee, Private Bag 4800, Christchurch (human.ethics@canterbury.ac.nz).
☐ I understand that if I want a summary of the results I can email the researchers.
☐ I consent to an audio recording of the interview being made and kept by the researcher.
☐ I understand that if I want copies of the audio recording and transcript of the interview I can email the researchers.
☐ By signing below, I agree to participate in this research project.

Name:

Signature:

Date:

Fig. 4. Consent Form


