Towards Automated Auditing for Network Configuration Changes

Yu-Wei Eric Sung, Sanjay Rao
Dept. of ECE, Purdue University
West Lafayette, IN, USA
{sungy,sanjay}@ecn.purdue.edu

Subhabrata Sen
AT&T Labs Research
Florham Park, NJ, USA
sen@research.att.com

Stephen Leggett
AT&T Inc.
Fishkill, NY, USA
sleggett@att.com

ABSTRACT

IP network operators face the challenge of making and managing low-level device configuration changes to accommodate the rapidly evolving needs of IP networks. As a first step towards automating this task, we explore a data-driven approach to audit high-level change activity in IP networks, by reverse engineering low-level longitudinal configuration data. We demonstrate the utility of our approach in auditing changes for five operational networks.

Categories and Subject Descriptors: C.2.3 [Network Operations]

General Terms: Management, Measurement

Keywords: Configuration changes, Network auditing automation

1. INTRODUCTION

One of the most challenging tasks for IP network operators involves managing changes to device configurations that are needed to reflect a change of network design, or as a response to troubleshooting problems in the enterprise network. This is daunting considering the large size and geographical span of enterprise networks, the diversity of configuration options, and the variety of devices. A small but incorrectly applied change can have a serious impact such as traffic being misclassified or blackholed, leading to adverse consequences such as SLA violations for the provider, and service disruptions for the customer enterprise.

Configuration changes are often *system-wide* in that they impact several devices, and *intertwined* in that executing a change may impact multiple parts of a configuration. Existing auditing tools are inadequate for two reasons. First, typical tools are geared towards managing and troubleshooting one device at a time. Second, changes are tracked at a low level (e.g., command logs), making it difficult to reason about the high-level semantics behind a change.

In this poster, we outline a methodology to reverse engineer high-level change activity in operational networks from router configuration files across time. Employing our methodology, we conduct a preliminary study of configuration changes in enterprise VPNs and demonstrate how the observed patterns enable change auditing. To our knowledge, this is one of the first works to study configurations *across time*. Our methodology employed a three-pronged approach as follows. Distinct from priori work, e.g., [1, 2], our work focuses on developing longitudinal views of changes, using a combination of data mining and domain knowledge.

(i) **Bottom-up Data Mining**: We used data mining techniques to uncover potentially important or anomalous change patterns since operator knowledge tends to be incomplete, and it is difficult for an operator to explicitly list all patterns of interest up-front. Further, there may be hidden patterns that an operator is unaware of.

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.
ACM 978-1-60558-175-0/08/08.

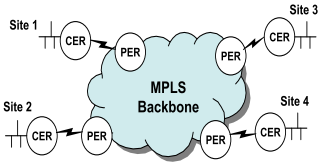

Figure 1: An example enterprise VPN topology.

Ent	City (Ctry,Ctint)	Network Growth	% New CERs	Config Size Min	Med	Max
E1	158(2,1)	-1.47%	8.21%	408	1033	1487
E2	100(31,5)	5.96%	16.56%	320	652	1175
E3	269(1,1)	25.2%	25.2%	551	633	1622
E4	162(36,5)	7.11%	25.26%	426	767	1475
E5	346(1,1)	-2.85%	1.66%	436	489	1104

Table 1: Enterprise information. The number of CERs ranges between 150 and 420.

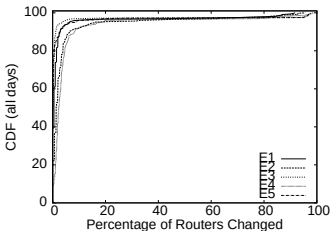

Figure 2: CDF of fraction of routers changed per day.

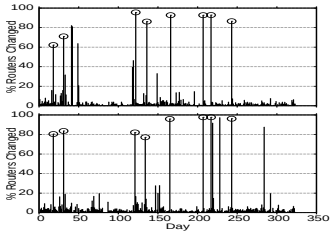

Figure 3: Fraction of routers changed over time for E2 (top) and E4 (bottom).

(ii) **Operator Domain Knowledge**: We corroborated uncovered patterns with operators since we believe an automated change auditing tools must evolve over time with operator-driven feedback so that only meaningful anomalies and correlations are highlighted.

(iii) **Logical Design Models**: To enable operators to better understand and reason about changes, we develop a logical model for Class of Service (CoS) configuration.

Enterprise VPN. Figure 1 shows an enterprise VPN spanning across multiple sites. Each site has a customer edge router (CER) connected via a WAN link to a provider edge router (PER). End-to-end CoS is provisioned by marking IP packets and treating packets differentially according to their markings, on the CER-PER-backbone-PER-CER path. We focus on changes to CERs, which are typically more dynamic and heterogeneous compared to provider routers. Our data set includes 11 months of daily archives of CER configuration for 5 enterprises as shown in Table 1. The networks exhibit diversity in size, geographic span, and growth.

2. GLOBAL SYSTEM-WIDE CHANGES

Configuration changes are sometimes *system-wide*, impacting most routers in a network. In this section, we seek to answer the following questions: (i) How frequently do changes have a system-wide impact? (ii) What operator objectives typically result in global changes? (iii) To what extent are anomalies present in global change events where a majority of routers, but not all, changed?

Change Frequency. Figure 2 depicts a CDF of the percentage of routers changed per day for all 5 networks, and Figure 3 shows the time series of the same metric for 2 of the networks. Across all networks, in 90% of the days, only 10% or fewer routers were changed, and in 3% of the days the impact was widespread - with more than 80% of the routers impacted. Note that some of these "global-event" days covered most, but not all, routers. Closer inspection revealed that these days were followed by the same change affecting the remaining routers in the subsequent days. Figure 3 also shows that many system-wide changes appear correlated (8 events circled in each sub-figure) across networks. This is consistent with the change practices, where the provider may schedule changes to several different enterprises in the same maintenance window.

Change Characterization. We identified a total of 51 global events, among which 15.4% and 84.6% were related to CoS and management or security operations, respectively.

Global Anomalies. We also want to understand why one or more outlier routers missed a global change. We divided the outliers in each global event (i.e., ⟨router-event⟩ pair) into *persistent* and *delayed*. Persistent outliers never received the missed update before the end of our data set whereas delayed outliers eventually saw the missed update. Table 2 summarizes the outliers. The numbers in parentheses denote the number of unique routers appearing as a particular outlier and the number of events in which some indicated outlier occurred.

Ent	Total	Persistent Outliers			Delayed Outliers
		errors	non-errors	unknown	
E1	172(38,8)	0	134(26,8)	3(3,2)	35(12,6)
E2	24(15,6)	11(6,3)	7(7,1)	5(5,3)	1(2,1)
E3	9(8,2)	0	2(2,2)	7(7,1)	0
E4	91(85,7)	0	81(78,2)	6(6,2)	4(4,3)
E5	16(3,6)	0	10(10,1)	0	6(3,4)

Table 2: Global outliers.

• **Persistent Outliers:** We classified persistent outliers into *errors* and *non-errors*. In a few cases, we were not able to determine the causes, and we classified them as *unknown*.

(i) *Errors*: These outliers were confirmed by the operators as needing fixes. We found 11 such outliers, all in E2. They corresponded to missed management updates, e.g., setting timeout for management sessions. The operators indicated that although best practice is to correct them, they were not critical to essential operations.

(ii) *Non-Errors*: These outliers were either confirmed or strongly suspected by the operators as genuinely not needing the update. As shown in Table 2, they constitute the majority of outliers detected in E1, E4, and E5. The 134 cases in E1 involve only 26 routers, all are related to an intentional CoS design. Non-errors in other networks were management-related. For example, low-end routers did not get the complete set of management ACL rules to reduce processing overhead. In addition, updates that increase parameter values (e.g., logging buffer size) above a threshold did not reach routers which already had them above the threshold.

• **Delayed Outliers:** Table 2 shows that E1 had the most delayed outliers, but on only 12 CERs. These routers used a special style of CoS configuration which required manual updates and were therefore not amenable to bulk updates effected through automated tools. In addition, while E2-E5 allow fixes to be made on-demand, E1 had a more stringent update process in that changes can be made only in pre-scheduled time windows. A combination of these two factors explains a large number of delayed outliers in E1. For E2-E5, one major cause of delayed outliers was that misconfiguration of a management ACL inadvertently blocked global updates, but was later discovered and fixed.

3. CORRELATION ANALYSIS OF CHANGES

In order to reflect complex higher-level objectives, configuration changes are often intertwined in that executing the change may require modifications to multiple routers or multiple blocks of a configuration. Operators often want to know what routers or configuration blocks tend to change frequently together to ensure consistency with policy objectives. In particular, we aim to answer the following questions: (i) How intertwined are changes within a configuration file and what higher-level objectives trigger the changes? (ii) How frequently do correlated changes occur? (iii) Do certain groups of routers exhibit similar change patterns?

Apriori Algorithm. We employed the *apriori algorithm*, a powerful data mining technique for association rule induction. It expresses an association between *items* within a *transaction*. We leverage it to find what tend to change together in a network. In our case, an association rule would be "If some access-list changes in a router, then in 50% of the time, some interface of the router changes, too." or "If routers x and y change together on a day, router z will always change on that day."

Correlated Changes. Our analysis uncovered syntactically unrelated relationships, such as updates to various server information due to management processes. Another interesting result was the correlations across routers located in geographic proximity to each other. For enterprise E5, our analysis revealed two groups of frequently changing configuration blocks corresponded to a major design change, one related to BGP routing, and the other related to removal of ISDN backup solutions. The two groups of updates were performed by independent design teams spanning a few months, configuring 20-30 sites every 2-3 days. The operators found our analysis useful in auditing that changes happened as intended.

4. LOGICAL-MODEL BASED ANALYSIS

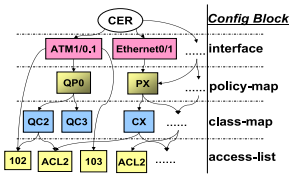

Figure 4: Intertwined CoS configuration.

To gain deeper insights on change activity, we also consider domain-specific analysis. We focus on CoS, a critical part of enterprise VPN configuration, which must accurately embed complex design choices, and misconfiguration can lead to SLA violations. Figure 4 shows the intertwined nature of CoS configuration in the Cisco IOS language. We built a change summarization tool that detects semantic changes by abstracting the CoS configuration as consisting of the number of supported classes, and the membership, policing parameters, and queuing parameters of each class.

CoS Anomalies: We discovered two types of CoS anomalies: (i) *no-impact changes:* intended changes shadowed by a preceding configuration line, which can potentially lead to misclassification of traffic; and (ii) *incomplete changes:* anomalies where two separate but related segments of configuration (i.e., policing and queueing rules) were not updated in a consistent fashion. These errors are usually fixed today in a *reactive* fashion, typically when they result in discernible performance degradation. A logical-model based auditing system can help fix these errors in a more proactive fashion, before the errors result in perceptible performance problems.

5. ONGOING WORK

The network operators have found our methodology to be very useful for understanding, detecting problems in, and troubleshooting change activity. We are currently developing a system for real-time profiling and auditing of configuration changes.

6. REFERENCES

[1] D. Caldwell, A. Gilbert, J. Gottlieb, A. Greenberg, G. Hjalmtysson, and J. Rexford. The cutting EDGE of IP router configuration. In *Proc. ACM SIGCOMM HotNet*, 2003.

[2] X. Chen, Z. M. Mao, and K. van der Merwe. Towards automated network management: Network operations using dynamic views. In *Proc. INM*, 2007.