Proposal on Network-Wide Rollback Scheme for Fast Recovery from Operator Errors

Kiyohito Yoshihara, Daisuke Arai, Akira Idoue, and Hiroki Horiuchi

KDDI R&D Laboratories Inc., 2-1-15 Ohara Fujimino-shi
Saitama 356-8502, JAPAN
{yosshy di-arai idoue hr-horiuchi}@kddilabs.jp

Abstract. This paper proposes a new network-wide rollback scheme for fast recovery from operator errors, toward the high availability of networks and services. A technical issue arises from the fact that operators, who manipulate one or more diverse devices and services due to their network-wide dependency in a typical management task, are the major cause of failure. The lack of systems or tools fully addressing the issue motivated us to develop a new scheme. The underlying idea is that, for any operational device or service, the observable behavior is identical whenever the same setting is configured. High availability will thus be achieved by rolling the settings that may cause an abnormal state by an operator error, back to past ones with which devices and services were stable. Certain policies for the network-wide rollback are identified and a prototype implementation and preliminary results will be presented.

1 Introduction

As seen from its global acceptance, the Internet is becoming as another form of promising infrastructure like electricity, water and gas. At the same time, the ever-increasing scale of the networks constituted by diverse devices and services entails additional opportunities for network operations. We operate one or more devices and services in a typical management task, due to their network-wide dependency. For instance, when we install a new Web server, we configure a router, the DNS server and a firewall in order, as well as the Web server.

In contrast, it is reported in [1] that operators are the leading cause of failure, accounting for 51%, which was roughly estimated based on the number of identified outages in three Internet sites of 500 to 5000 computers. In order to make the networks and services dependable enough to be used with availability equivalent to existing infrastructures, the new technical issue arising is to enable fast recovery from operator errors while considering the diversity of devices and services as well as their network-wide dependency.

Certain systems [2, 3] have been developed to address the issue, which presuppose errors on the part of the operator, provide undo utilities that allow the operator to roll a service back to a previous state, and minimize the Mean Time To Repair (MTTR) for higher availability; however, the systems take into account neither the diversity of services nor their dependency. The development

of additional software components for each service is required for the adoption. Moreover, their operational scope is restricted to a single service, and the resulting state is not necessarily consistent with that after past management tasks. We might exploit tools [4, 5] for multi-vendor routers and switches; however, they are the same as the systems [2, 3] in terms of the limited operational scope.

This paper proposes a new network-wide rollback scheme for fast recovery from operator errors, as work in progress. The proposed scheme will be differentiated by (1) the practicality by which we can deploy the proposed scheme with no modification to existing diverse devices and services and (2) the network-wide configuration management, via which we can roll an entire managed network back to a safe state, consistently with past management tasks. In the subsequent sections, we will initially present the proposed scheme, together with some policies specific to network-wide rollback, and then show the current state of the prototype implementation and preliminary results.

2 Proposal on Network-Wide Rollback Scheme for Fast Recovery from Operator Errors

2.1 Design Principle

The idea of the proposed scheme comes from a straightforward fact: for any operational device or service, its observable behavior is identical whenever the same setting is configured. Thus, we can expect fast recovery from operator errors when in an abnormal state after a management task, by rolling the current settings back to past ones, with which managed devices and services were in a stable state. We extend this idea to a network with the design principles below.

1. We develop a new server to achieve network-wide rollback. The server hooks all command requests to and responses from managed devices and services during a management task, and backups all settings of the devices and services, including those that were not operated whenever a task is completed. An ID is given, along with backup time and an associable comment, in order to identify a state in the network-wide rollback.
2. For a device and service, we describe two scripts for the backup and rollback of setting, and register them as well as the human-readable name, IP address and login credential with the server. Running one or more scripts can see a network of diverse devices and services rolled back to a stable state, with no modification to the devices and services.
3. To cope with addition/deletion of a device and service, we define compensation policies as retaining a state stably after the network-wide rollback over such changes. A simple example of the policy is to skip the rollback for a device not existed before. The resulting state should be consistent but may not always be the one occurred in the past management tasks.
4. We cannot always roll all necessary settings back at will, due to a transient state during the rollback: e.g. inability to reach a managed device a few hops away from the server, via the change of a routing table of a router on the path. With this in mind, we define ordering policies to avoid such side-effects.

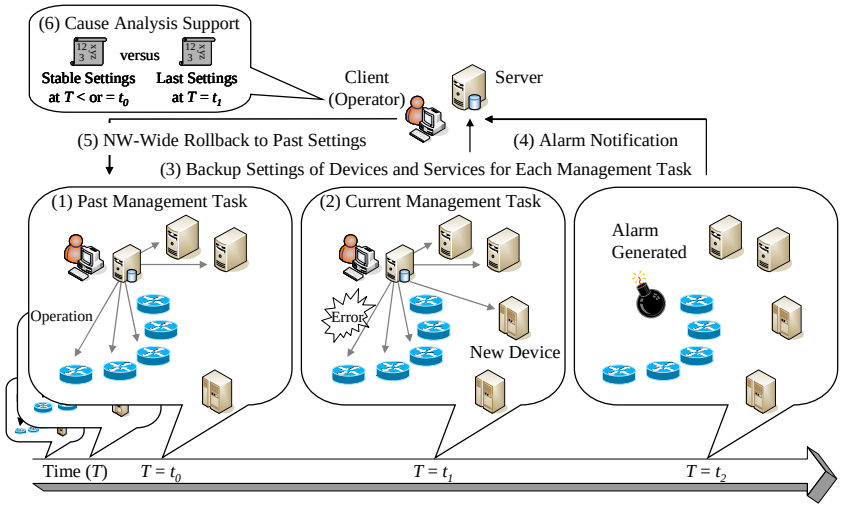

Fig. 1. Overview of System Operation Based on Proposed Scheme

2.2 Overview of System Operation Based on Proposed Scheme

We assume below that the enrollment of names, IP addresses, login credentials, and scripts of managed devices and services in the server has been completed.

When an operator performs a management task, he/she firstly logs into the server, and subsequently further logs into a target device and service from the server, via a usual terminal client (Fig.1 (1) and (2)). Every time the operator logs out from the device and service, the server backups all the settings of the managed devices and services, using the scripts (Fig.1 (3)). The backup can be on a regular basis and on demand from an operator.

In case of an alarm notification (Fig.1 (4)) from an external system, suspecting operator errors in the last management task (Fig.1 (2)), an operator logs into the server for fast recovery. Based on compensation and ordering policies, he/she then rolls all settings of the managed devices and services back (network-wide rollback) to the past ones in a stable state at $T \leq t_0$ (Fig.1 (5)), from the server.

Subsequently, an operator will conduct a cause analysis of the alarm. The proposed scheme supports the comparison of the last settings at $T = t_1$ with the others at $T \leq t_0$, and the verification of the commands hooked in the past management task (Fig.1 (6)). When the alarm is still alive after the network-wide rollback, there might be other causes, such as hardware failure, or software crash unable to reboot. Coping with the causes of the latter kind is future work.

3 Implementation and Preliminary Results

Figure 2 shows the prototype implementation for the evaluation of the proposed scheme. The development of the policy DB is underway.

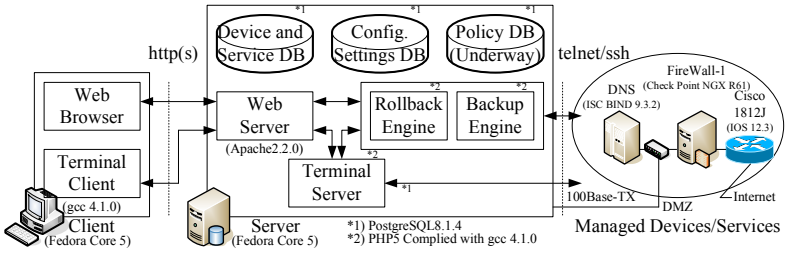

Fig. 2. Prototype Implementation Based on Proposed Scheme

The primary performance metric is the time of the network-wide rollback, for the faster it is completed, the shorter MTTR for high availability is accomplished. We applied the proposed scheme to the network of a router, a DNS server and a firewall assuming a barebones DMZ, as shown on the right side of Fig.2. The time was 142 sec., when they were sequentially rolled back accompanied with reboots, providing an upper bound of the rollback time for the network. We could reduce the time to 40 sec. by starting the rollback of the DNS server earlier than that of the router, followed by the router rollback soon after that. In the case of inverse order, the rollback of the DNS server failed until completion of the router rollback, due to the inability to reach its higher level DNS servers.

The above preliminary results reveal that the rollback time depends on an ordering policy. We are investigating compensation and ordering policies, and faster rollback operations that do not rely on time-consuming reboots. We will evaluate them in a carrier-scale network, using the above bound as a reference.

4 Conclusions

This paper proposed a new network-wide rollback scheme for fast recovery from operator errors, as a work in progress. The design principles and an overview of the system operation were presented. We will show the evaluation results shortly, using the completed version of the prototype implementation.

References

1. Patterson, D.A.: A Simple Way to Estimate the Cost of Downtime. In: Proc. of the 16th Systems Administration Conference. (November 2002) 185–188
2. Brown, A.B., Patterson, D.A.: Undo for Operators: Building an Undoable E-mail Store. In: Proc. of USENIX 2003. (June 2003) 1–14
3. O'Brien, J., Shapiro, M.: Undo for anyone, anywhere, anytime. In: Proc. of the 11th workshop on ACM SIGOPS European workshop. (September 2004)
4. Shrubbery Networks, Inc.: Really Awesome New Cisco confIg Differ (RANCID). `http://www.shrubbery.net/rancid/` (URL available for May, 2007).
5. AdventNet, Inc.: DeviceExpert. `http://manageengine.adventnet.com/products/device-expert/index.html` (URL available for May, 2007).