

NETWORK TRANSFORMATION

Isolating, Transforming and Deploying Bulk Policies

Network transformation is the large-scale modification of a network for various business reasons. Examples abound, and the common reasons are large scale application migration in a data center, improving network security through increased segmentation, device consolidation in the context of a hardware upgrade, virtualization of application hosts, renaming objects and re-mapping IP addresses to impose corporate conventions, and consolidating disparate networks arising out of a corporate merger. This paper captures some of the challenges that are involved in the implementation of a network transformation project and how engineering automation technology can significantly help in reducing the effort and risk involved.

1. **Bulk policy change:** Most network transformation projects involve massive policy changes across many network devices. Allowing a service to a migrated application involves changes in several devices along a path that route packets to the application server. With several such routes to be changed, the likelihood of errors increases. To manage this process, scripts in vendor-specific format, must be uploaded and run directly on the network devices or on a vendor-supplied console. With engineering automation, the creation of these scripts can be automated to ensure that mass changes that are deployed are correct.
2. **Can't deploy changes in one change window:** It is not possible, nor advisable, to implement and deploy all transformation changes in one change window. Transformation projects are often divided into multiple change windows with each window targeting a related group of application servers. This requires a great deal of analysis to identify the applications that will be migrated in a change window, and planning to ensure that staggered change windows do not endanger business continuity due to the dependencies between the applications. The automated generation of deployment scripts must incorporate knowledge of application dependency and changes made in prior change windows.
3. **Identifying and isolating policies:** Most network transformation projects do not have readily identifiable policy changes that engineers can deploy. As described above, the project is divided into smaller projects and a migration plan created. A sound engineering migration plan consists of a time line of planned change windows, with a description of the applications, the hosts, the supporting services that must be enabled, and the firewall and router policies that need to be migrated to the target network in each change window. This requires identifying and isolating existing policies in routers, firewalls and application servers before performing the transformation. Without automation, the process is not repeatable across the entire project, leading to very uneven outcomes. It is also highly inefficient. As a result, some applications maybe migrated flawlessly; others may cause a business outage with the culprit not being the migrated application or its data, but incorrect or inadequate changes in network policy.

In some instances, a list of applications, and their services are used to isolate policies that enable these services from various clients. Object groups and hierarchies make the manual isolation process very hard, especially when there are a large number of rules and objects.

In other transformation projects (like when a flat application tier architecture is being transformed into a multi-tier architecture with multiple zones separated by firewalls), new policies that respect the segmentation may have to be created. The basis for this is traffic that was flowing to the application in the old network. Packet trace and application discovery tools have to be used to first capture the policy that is currently active, isolate the policy that is being moved by specific applications, and transform it before

deployment. This process requires significant collaboration with application, network, and other IT stake holders.

In many cases, policies that are identified for migration are too broad and may also contain obsolete policies that have been left over from a previous time. Removing the unused policies and narrowing the broad rules is essential before starting the transformation project. This will avoid carrying over clutter to the target environment that might already be complex on its own. Firewall log or packet flow data can guide this cleanup and policy narrowing process.

Querying facilities that identify all rules and objects by searching across object hierarchies using names, IP addresses, protocols, and ports is essential for isolating required rules that need to be moved. The ability to browse the identified rules and objects, remove unused or duplicative policies, and placing them into a staging area, or exporting them using reports are important initial steps leading to automated network transformation script generation.

- 4. Policy transformation:** Policy transformations are many and are typical of network transformation tasks to which engineering automation can be applied to a high degree. These include re-mapping IP addresses, changing address translations, renaming objects to conform to target network naming conventions or creating new objects. When the target firewall platform is from another vendor, differences in architecture must be taken into account. Examples include adding source and destination zones for the policies in a Netscreen firewall or access groups for the ACLs in a Cisco firewall.

When the policies have to be merged into an existing target firewall, rule dependencies need to be taken into account and should be placed in the proper rule position for the policy to take effect. Rules that are already covered by existing rules in the target firewall should be removed to prevent clutter. Wherever possible, the rules being merged should be incorporated into existing rules to keep the rule base small. Object duplication and name conflicts should be avoided when objects are added into the target firewall.

- 5. Production systems are not static:** Deploying policies on production firewalls or routers must take into account policy deltas. These are changes in policy that have been put into production since the start of policy transformation, which was based on an earlier snapshot. Engineering automation makes it possible to reduce the size of the deltas, by allowing the snapshot to be delayed as much as possible to make it close to the change window. Nevertheless deltas, albeit reduced, still remain. The production change window may be tightly controlled and also very small for completing the changes. Automation can rapidly evaluate the rule and object changes in the reduced delta and create the incremental policy scripts necessary.
- 6. Cataloging changes and verification:** To mitigate risk and guarantee the success of the transformation project, it is essential to document the changes, and validate the correctness and completeness of the changes. Analyzing the impact of those changes on traffic flow will greatly help in catching any errors before the changes are deployed to the production system and thereby reduce the duration of live application network access testing. Offline analysis also reduces significant post-production application testing and errors in the firewall policies can be identified without injecting actual packets. Application testing, which is under the control of a separate application team, is still required to make sure that the applications are set up and running properly on the appropriate application servers.
- 7. Creating revert scripts to back out the changes:** In the event that something goes wrong on the production firewalls, changes may need to be backed out. If the management servers provide the capability to save and revert to an earlier configuration completely, then there is not an issue. Otherwise, a revert script must be readily available. A revert script is the mirror image of the transformation script discussed above. A revert script is generated from the transformation script, so it reverses only those changes the transformation script made. Most management servers provide a facility to back up the rule bases but it is not as clear for global objects, as in the case of Check Point Provider-1 Customer Management Add-ons.

For Complex Network Transformation Projects:

Athena's licensable tools can be used in a variety of scenarios such as moving applications across data centers, creating a layered segmented network to implement security-in-depth, re-architecting network access policies by splitting or combining individual firewall policies, re-mapping of IP addresses, or performing platform migrations.

In all of these scenarios we provide significant automation to isolate the relevant rules, facilitate the mapping process, cleanup structural redundancies, generate scripts and validate new policies on target devices, including network path validation.

To learn more about our solutions for Network Transformation contact:

Athena Security

sales@athenasecurity.net

1-877-339-0215

<http://www.athenasecurity.net>

About Athena Security

Athena Security, Inc. offers infrastructure analysis tools that identify the precise relationship between firewall rules and network services in a single device or across a complex network. With a comprehensive focus on configuration data, Athena helps network and security engineers perform what-if analysis that reduces the reliance on diagnostics and validation by testing. Over 300 companies turn to Athena products, Athena FirePAC™, Athena PathFinder™, and AthenaVerify™, for standardized and consistent intelligence to reduce the time and effort required for policy management on network security devices. For more information see <http://www.athenasecurity.net/>.