

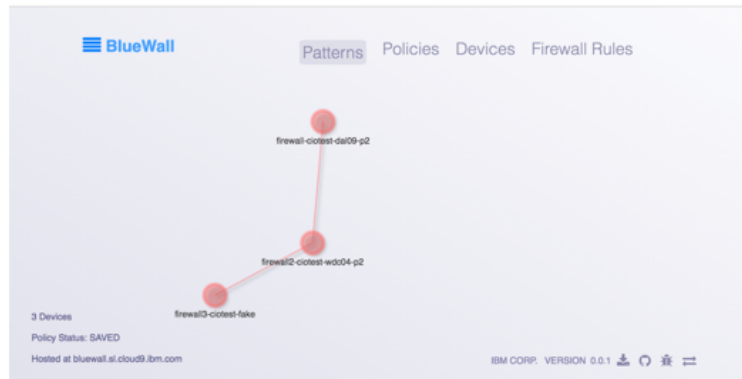
BlueWall
Software Defined Network Management
in Hybrid Enterprise Cloud Environments

Jinho Hwang, Jin Xiao, Nikos Anerousis
{jinho, jinoaix, nikos}@us.ibm.com

BlueWall is a service for distributed network management in hybrid cloud environments, specifically including on-premise data centers, Infrastructure as a Service (SoftLayer, AWS), and Platform as a Service (Bluemix). While the current centralized network management require exposures of the environments to the centralized management stack, the BlueWall only needs to deploy one micro-service into each environment and secure communication channel back to the orchestrator. This distributed approach allows different policies and application patterns be governed in different network environments, which enables high flexibility when deploying cloud instances in hybrid cloud platforms. We show how BlueWall orchestrates different network environments in a distributed manner, and further how different policies are applied. Finally, we discuss lessons learnt and challenges encountered during this hybrid network management.

Outline

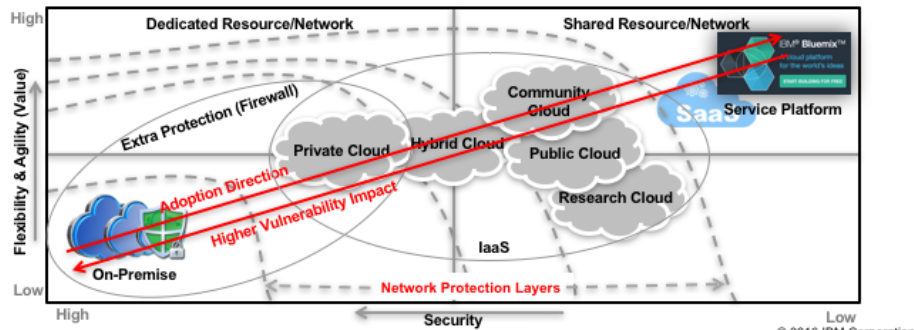
- Background and Motivation
- Hybrid Cloud Network Management
- Lessons Learnt
- Related Work
- Conclusion



We explain background and motivation of the hybrid cloud network management, and discuss challenges, benefits of distributed network management, and our approach. We illustrate the BlueWall architecture and deployment model. Lastly, we discuss lessons learnt.

Hybrid Clouds and Network Management

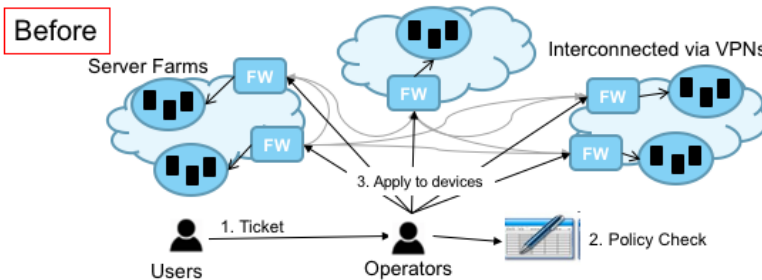
- Hybrid challenges
 - Dealing with heterogeneous infrastructure and network technologies without necessarily a standard API or function set
 - Dealing with different modes of access management
 - Dealing with different security policies, security domains, and devices
- Support on-demand networking
 - Automate network security configurations is a must to achieve consistency and compliance
 - Important to streamline security approval processes and adhere to system of records



Hybrid cloud holds tremendous promise to enterprises as it allows the enterprise to flexibly deploy services and workloads among on premise, public data centers and private data centers. In many industry market sectors (e.g., banking, healthcare, etc.), hybrid cloud is the only method for enterprises to establish a cloud-centric infrastructure whereby sensitive data are only stored and processed on premise. It follows then hybrid cloud management solutions are needed to address the three listed issues. Each cloud environment today has its own infrastructure design, service mix and methods of access, deploy and operation, especially for network provisioning, configuration and security management. Security policies and the definition of security domains are also domain and infrastructure dependent, that gets further complicated and widened as workloads are spread across multiple cloud infrastructure. At the same time, with increasing automation and resource virtualization, on-demand service/application creation and deployment is a common strategy used to support agile business and development models, and network on-demand challenges us to provide solutions for rapid security configurations that are expedient, consistent and compliant across hybrid environments. Alongside this automation, is the need for streamlined security approval process that can support ease of human administration and compliance audits.

Motivation

- **Server provisioning takes several minutes, but validating/opening firewall rules between security zones takes several days or even weeks**
 - Layers of groups need to validate the requested firewall rules are secure, even with the repetitive ones
 - After the approval, network engineers need to put the rules into the network appliances
- **Security policies and business logics are not encoded to be consumed**
 - Holistic network information (such as subnet – security zone, etc.) is not consolidated anywhere. Network engineers keep them in their own spreadsheets or brain
 - Security policies and business logics need to be encoded
- **Firewalls are core network security elements of a target solution but hard to get right**
 - Steady-state firewall managements has been done by human labor only
 - Large number of interacting firewall rules in distributed firewalls increases significantly the possibility of target configuration errors and network vulnerabilities
 - Firewall automation (incl. garbage collection) is necessary



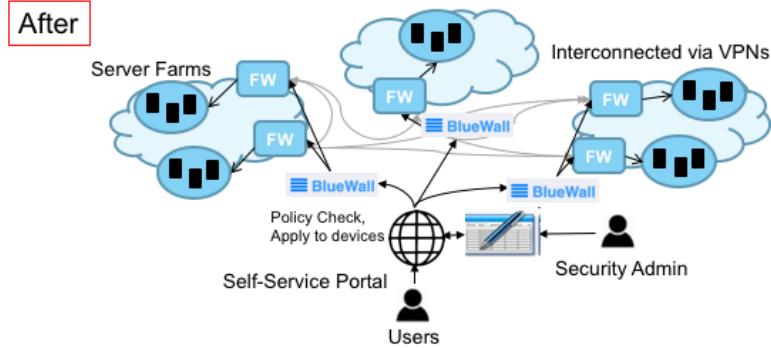
4

© 2016 IBM Corporation

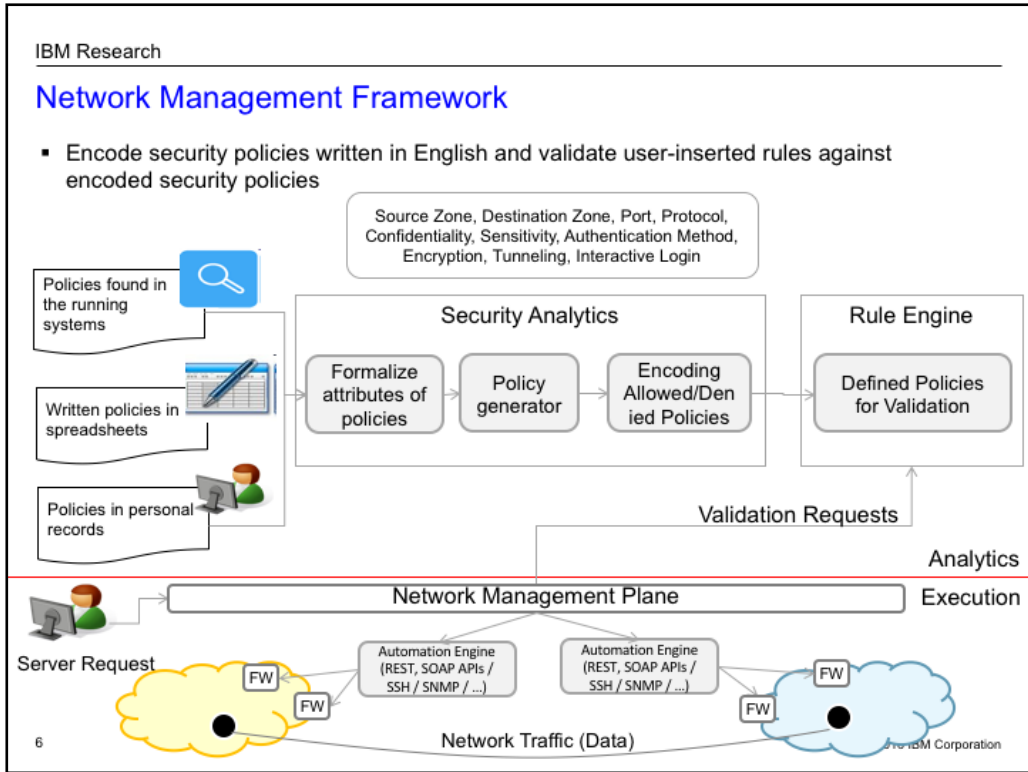
The network security is one of the most important measurement of enterprises. This becomes even more important when enterprises are moving to the cloud environments. Instead of moving everything into the cloud, many companies choose to run hybrid cloud modes with combination of on-premise data centers, public data centers, and private center centers. With the automation of clouds, server provisioning takes several minutes, but validating/opening firewall rules between security zones takes several days or even weeks with stringent security policies, often written in the documents and examined by the security administrators. The main reason for this slow process is that security policies and business logics are not usually encoded to be consumed, and holistic network information (such as subnet – security zone, etc.) is not consolidated anywhere. Also, without the right automation in place, enterprises are wary of large number of interacting firewall rules in distributed firewalls increasing significantly the possibility of target configuration errors and network vulnerabilities.

Technical Approach

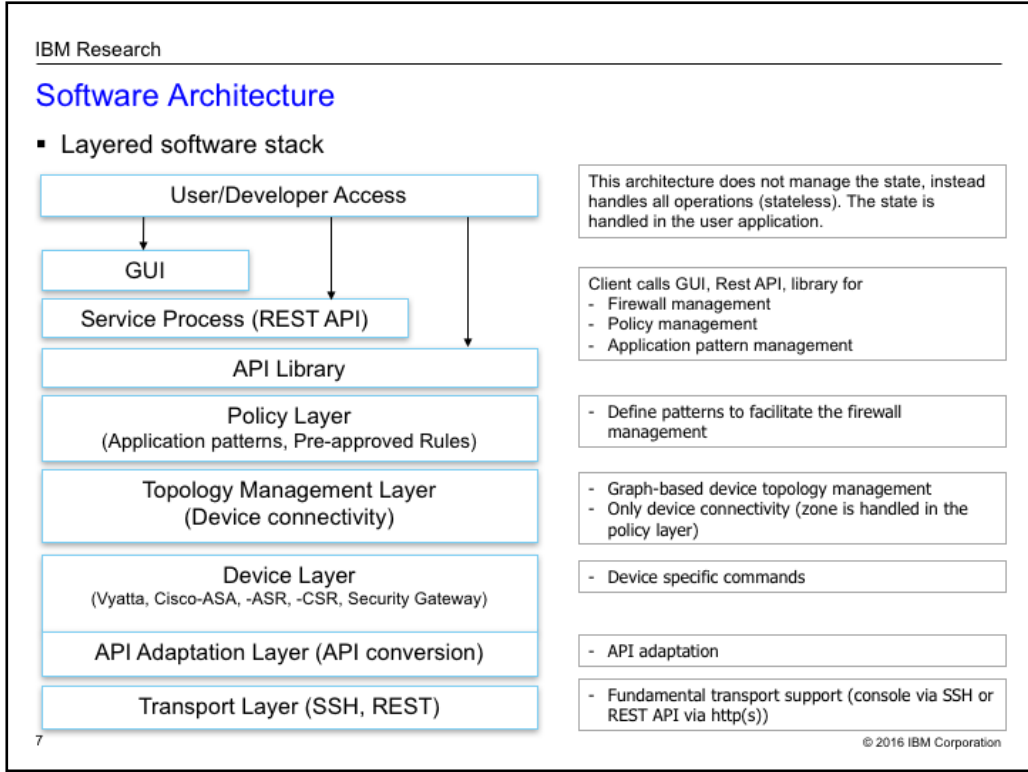
- Self-service for managing firewall request when servers (or services) are created (via APIs)
- Human tasks for end-to-end network management process including policy validation, rule implementation are automated in the backend
- Supporting scalable network management (distributed network management)
- Handling different types of devices
- Network management targets: IaaS, PaaS, SaaS



In order to provide a correct network configuration consistent and compliant to the security policies, the automation, audit, and compliance are required. The central orchestrator only keeps the default security policies and application patterns associated with them, but all the automation that manipulate the devices and environmental policies are done in each cloud environment level. To achieve this level of automation, a self-service for managing firewall request when servers (or services) are created (via APIs), and human tasks for end-to-end network management process including policy validation, rule implementation are automated in the backend. Also, this distributed network management supports scalable network management, and handles different types of devices.



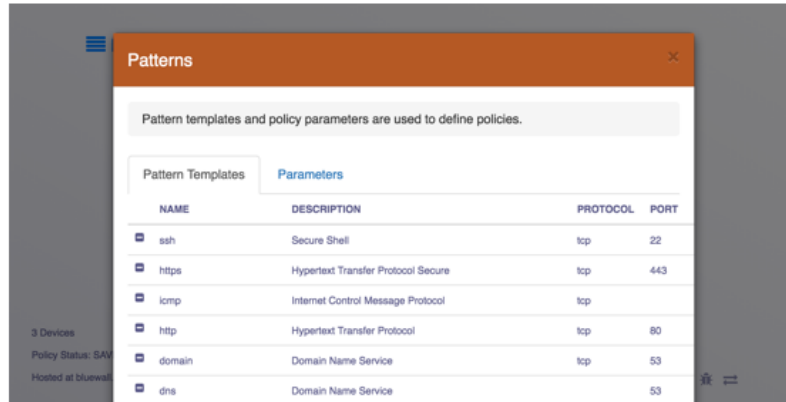
The network management (orchestration) plane framework includes the migration of existing security policies into the consumable (encoded) security policies. The identified security attributes include source zone, destination zone, port, protocol, confidentiality information, sensitive personal information, authentication method, encryption, tunneling, and interactive login. Policies are mined from the written security documents, running systems, and personal records, and analyzed to draw common security policies. Then, the defined policies are used for the firewall rule validation in real time when users request new firewall rules and compliance checks are done.



Each micro-service deployed into each cloud environment has this software architecture that covers security policies, topology managements, and device access layers. The network orchestrator access each micro-service through REST APIs, and administrators can also access the service via a graphical user interface. The policy layer downloads the security properties and required rules from the network orchestrator and can append new policies based on the environment. The topology management layer keeps track of the device connectivity in order to find a flow path. The device layer allows this service to access devices via ssh, SOAP, or REST APIs.

Application Pattern

- Application patterns are captured in order to collect pre-approved patterns
- Patterns are also used to automatically populate information to users' requests
- Properties include protocol, port, authentication, confidential information, sensitive personal information, encrypted, tunnel, interactive login, scope



Pattern templates and policy parameters are used to define policies.

Pattern Templates Parameters

NAME	DESCRIPTION	PROTOCOL	PORT
ssh	Secure Shell	tcp	22
https	Hypertext Transfer Protocol Secure	tcp	443
icmp	Internet Control Message Protocol	tcp	
http	Hypertext Transfer Protocol	tcp	80
domain	Domain Name Service	tcp	53
dns	Domain Name Service		53

3 Devices
Policy Status: SAN
Hosted at bluewall

The application patterns are used to collect pre-approved application patterns that can automatically populate the corresponding properties to certain application. The application patterns allow users to specify the application name in order to find all the pertinent properties including protocol, port, authentication, confidential information, sensitive personal information, encrypted, tunnel, interactive login, scope.

Security Policy

- Network security policies (i.e., zone policy) are captured to validate the firewall rule requests
- Network policies can be adjusted depending on how cloud environments are set up
- Each network policy includes a list of application patterns and their properties



The network security policies include zones and devices that are used to validate the firewall rule requests. The minimally required security policies are downloaded from the network orchestration plane. Depending on the flow path from which device/zone to device/zone, an applied policy is determined and validated against it. Network policies can be adjusted depending on how cloud environments are set up, and Each network policy includes a list of application patterns and their properties.

Self-Service Firewall Rule Request

- Self-service firewall request page allows users to put some custom information
- Users can immediately validate the requesting rules are approved or not
- Application patterns are used to populate security properties
- Security policies are also shown to match the exact location of requested IPs

The screenshot shows a web form titled "Firewall Rule" with a close button in the top right. Below the title is a grey box with the text: "A firewall rule can be validated and applied using pre-defined patterns and policies." The form is divided into sections:

- CONNECTION**: "Provide source and destination addresses." Fields for SOURCE IP (10.143.48.2) and DESTINATION IP (10.175.10.129) are present, with a "FIND FLOWS" button below.
- POLICY**: "The rule will be validated against the following policy." Field for POLICY NAME (gr2protys).
- PATTERN**: "Choose a pattern that is already approved or create a new rule." Includes a "Type" dropdown and an "OPTIONAL FIELDS" section.
- OPTIONAL RULE**: "These parameters override the pattern (you can skip this)." Fields include: DESCRIPTION (bluewall generated rule), PROTOCOL (tcp), PORT (443,8443), AUTHENTICATION (key,certificate), CONFIDENTIAL_INFORMATION (false), SENSITIVE_PERSONAL_INFORMATION (false), ENCRYPTED (true), TUNNEL (true), and INTERACTIVE_LOGIN (true).

At the bottom, there is a red "Verify Rule" button and the text "ACCEPTED". A "Close" button and a blue "Submit" button are at the bottom right.

Users can submit the firewall rule requests through the self-service firewall request form. Users can immediately validate the requesting rules, which used to take a couple of days or weeks going through the human approval chains. Also, applications patterns are used to populate security properties for the sake of consistency and convenience. Security policies are also shown to match the exact location of requested IPs.

Devices (Topology) Management

- Each BlueWall instance manages firewall devices under its jurisdiction
- When firewall rules are requested, the network flow paths are found automatically
- New devices can be easily added

The screenshot displays the BlueWall management interface. On the left, a network topology diagram shows three firewall devices connected in a triangle: 'firewall-ciotest-dal09-p2' at the top, 'firewall2-ciotest-wd04-p2' at the bottom right, and 'firewall3-ciotest-lake' at the bottom left. On the right, a 'Devices' management window is open, showing a table of device specifications and connection options.

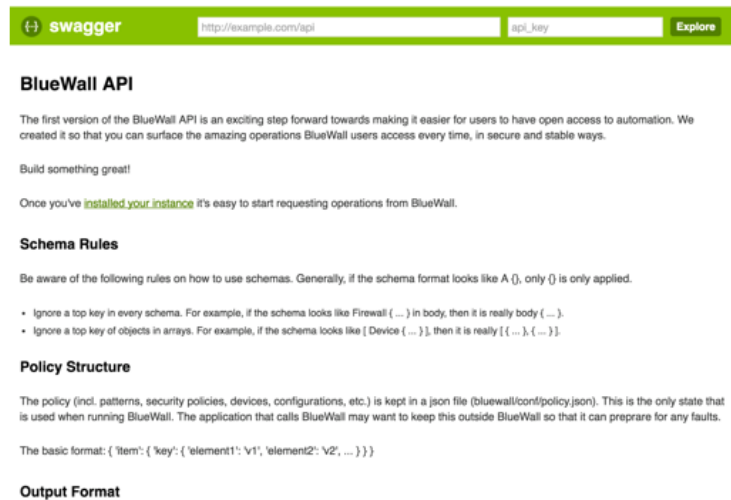
NAME	TYPE	USERNAME	PASSKEY	ADDRESS	VLANS
firewall-ciotest-dal09-p2	vyatta	jinho		169.55.238.140	[[{"vlanid":"2115","policy":"dmz"}]]
firewall2-ciotest-wd04-p2	vyatta	jinho		169.45.206.11	[[{"vlanid":"7911","policy":"dmz"}]]
firewall3-ciotest-lake	vyatta	jinho		169.45.206.12	[[{"vlanid":"17911","policy":"dmz"}]]

Below the table, there is an 'ADD MORE DEVICES' button and 'Close', 'Commit', and 'Save' buttons. The interface also shows '3 Devices' and 'Policy Status: SAV'.

The devices are managed in each service of each environment. The device information is used to discover the flow path along the network routing path, and this helps to determine which security policy needs to be used across different devices (and zones). The network orchestration layer does not need to know anything related to the devices.

Swagger APIs

- South-bound APIs are provided in Swagger
- Basic, Session, API key based authentications are supported



The screenshot shows a Swagger API interface. At the top, there is a green header with the Swagger logo, a text input field containing 'http://example.com/api', another text input field labeled 'api_key', and an 'Explore' button. Below the header, the title 'BlueWall API' is displayed. The main content area contains several sections: a paragraph describing the API, a 'Build something great!' section, a 'Schema Rules' section with two bullet points, a 'Policy Structure' section with a paragraph and a code snippet, and an 'Output Format' section.

BlueWall API

The first version of the BlueWall API is an exciting step forward towards making it easier for users to have open access to automation. We created it so that you can surface the amazing operations BlueWall users access every time, in secure and stable ways.

Build something great!

Once you've [installed your instance](#) it's easy to start requesting operations from BlueWall.

Schema Rules

Be aware of the following rules on how to use schemas. Generally, if the schema format looks like A {}, only {} is only applied.

- Ignore a top key in every schema. For example, if the schema looks like Firewall { ... } in body, then it is really body { ... }.
- Ignore a top key of objects in arrays. For example, if the schema looks like [Device { ... }], then it is really [{ ... }, { ... }]

Policy Structure

The policy (incl. patterns, security policies, devices, configurations, etc.) is kept in a json file (bluewall/conf/policy.json). This is the only state that is used when running BlueWall. The application that calls BlueWall may want to keep this outside BlueWall so that it can prepare for any faults.

The basic format: { 'item': { 'key': { 'element1': 'v1', 'element2': 'v2', ... } } }

Output Format

The south-bound APIs are provided in the swagger format. Currently, the BlueWall provides 5 APIs including policy, firewall, verify, flows, and patterns. The policy APIs allow administrators or orchestrators to update the security policies. The firewall APIs allow users to request firewall rules. The verify APIs enable real time rule validation. The flows discover flow paths with source and destination pair. The patterns provide pre-defined application patterns. The authentication method include basic, session, and key based approaches.

Related Work

▪ Academia

- Van Tran, Jacky Keung, Anna Liu, and Alan Fekete. 2011. Application migration to cloud: a taxonomy of critical factors. In *Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing (SECloud '11)*. ACM, New York, NY, USA, 22-28. DOI=<http://dx.doi.org/10.1145/1985500.1985505>
- P. V. Beserra, A. Camara, R. Ximenes, A. B. Albuquerque and N. C. Mendonça, "Cloudstep: A step-by-step decision process to support legacy application migration to the cloud," *2012 IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA)*, Trnto, 2012, pp. 7-16.
- X. Meng, J. Shi, X. Liu, H. Liu and L. Wang, "Legacy Application Migration to Cloud," *Cloud Computing (CLOUD)*, 2011 *IEEE International Conference on*, Washington, DC, 2011, pp. 750-751.
- B. C. Tak and C. Tang, "AppCloak: Rapid Migration of Legacy Applications into Cloud," *2014 IEEE 7th International Conference on Cloud Computing*, Anchorage, AK, 2014, pp. 810-817.
- M. Nidd, K. Bai, J. Hwang, M. Vukovic and M. Tacci, "Automated business application discovery," *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, ON, 2015, pp. 794-797.
- Dannver Wu, Jinho Hwang, Maja Vukovic, Nikos Anerousis, "BlueSight: Automated Discovery Service for Cloud Migration of Enterprises," *International Conference on Service Oriented Computing (ICSOC)*, Demo Track, 2016

▪ Industry

- Cisco Network Services Orchestrator (NSO), <http://www.cisco.com/c/en/us/solutions/service-provider/solutions-cloud-providers/network-services-orchestrator-solutions.html>

Van Tran, Jacky Keung, Anna Liu, and Alan Fekete. 2011. Application migration to cloud: a taxonomy of critical factors. In *Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing (SECloud '11)*. ACM, New York, NY, USA, 22-28. DOI=<http://dx.doi.org/10.1145/1985500.1985505>

P. V. Beserra, A. Camara, R. Ximenes, A. B. Albuquerque and N. C. Mendonça, "Cloudstep: A step-by-step decision process to support legacy application migration to the cloud," *2012 IEEE 6th International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA)*, Trnto, 2012, pp. 7-16.

X. Meng, J. Shi, X. Liu, H. Liu and L. Wang, "Legacy Application Migration to Cloud," *Cloud Computing (CLOUD)*, 2011 *IEEE International Conference on*, Washington, DC, 2011, pp. 750-751.

B. C. Tak and C. Tang, "AppCloak: Rapid Migration of Legacy Applications into Cloud," *2014 IEEE 7th International Conference on Cloud Computing*, Anchorage, AK, 2014, pp. 810-817.

M. Nidd, K. Bai, J. Hwang, M. Vukovic and M. Tacci, "Automated business application discovery," *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ottawa, ON, 2015, pp. 794-797.

Dannver Wu, Jinho Hwang, Maja Vukovic, Nikos Anerousis, "BlueSight: Automated Discovery Service for Cloud Migration of Enterprises," *International Conference on Service Oriented Computing (ICSOC)*, Demo Track, 2016

Cisco Network Services Orchestrator (NSO), <http://www.cisco.com/c/en/us/solutions/service-provider/solutions-cloud-providers/network-services-orchestrator-solutions.html>

Lessons Learnt

- Focus on north-bound management interfaces and functions
- Facilitate and simplify policy specification and policy-to-configuration mapping requires structured and technology-assisted process management, and codified security zone and rules classification
- Support open APIs to embrace inter-domain corporation, ease management integration, and function reusability
- Important to keep human in the loop for both security safeguard and compliance, means providing technology assistance in automating repetitive tasks, provide informative and comprehensive systems of records, and provide scalability through patterns

Through our research and development, we've come to learn many voids need to be filled to achieve hybrid cloud management. In particular, hybrid cloud network security management and automation requires a comprehensive encapsulation, definition and representation of security policies, patterns, topology and domain specific administration rules. Network automation needs to be supported by a overarching orchestrator and governed by a security policy approval process. In order to facilitate and simplify policy specification to achieve management scale, process management need to provide security zone abstraction, pattern-based specification of policy rule sets, and dealing with inter-zone policy specification in broad strokes, while providing consistent mapping of policy to rules. We also adhered to open API designs which is much needed in developing hybrid cloud management solutions as it allows for easy integration of new policies and technologies in a transparent and standardized way, and promise cross-domain function reuse. Finally, it is important to keep human in the loop, as humans are ultimately accountable for both security safeguard and compliance audits. We focused on technology assistance that automate repetitive and mundane tasks at the rule creation and network configuration level, while keeping human administration at the top policy and inter-domain level by providing abstraction, domain association and end-to-end workflow view across hybrid infrastructure. We further provided detailed and consistent systems of records that enabling tracking any network configuration requests among its requesting application/service, applicable security policies across security zones, approval records, and log of network execution.

Conclusion and Future Work

- Presented BlueWall: Software Defined Network Management in Hybrid Enterprise Cloud Environments for managing firewall request when servers (or services) are created (via APIs).
- Demonstrated hybrid network management design and self-service capabilities.
- Discussed challenges arising in network management in the hybrid enterprise cloud environments.
- Future Work
 - Analytics capabilities to continuously guarantee the network compliance:
 - Rule optimization by combining and segmenting into blocks
 - Compliance checking through the auditability
 - Network monitoring analytics for trouble shootings and diagnosis

We presented BlueWall -- a service for distributed cloud network management in hybrid cloud environments. We showed how BlueWall manages security policies, application patterns, device topologies, how BlueWall orchestrates different network environments in a distributed manner. We also discussed lessons learnt and challenges encountered during this hybrid network management.

Currently our system focuses on automation of the firewall validation and automation of the rule implementation in the hybrid environments spanning different cloud data centers. Our future work is two-fold, we plan to focus on automated network audit and compliance checks of existing rules, especially when policies are changed. Secondly, we want to extend the platform to support the network analytics that can monitor the network devices and perform trouble shootings. That will require distributed analytics processing platform, and consolidated data store for the comprehensive analytics.