ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

The Scenario of Enhancing Network Security Through Software Defined Networking (SDN)

R.Shanthi Prabha, Dr Rs.Vetrivel
Assistant Professor, Dept. of Computer Science,
Sri Adi Chunchanagiri Women's College, Cumbum, Theni(dt).
Professor, Computer Science, Subramanya College of Arts and Science, Palani

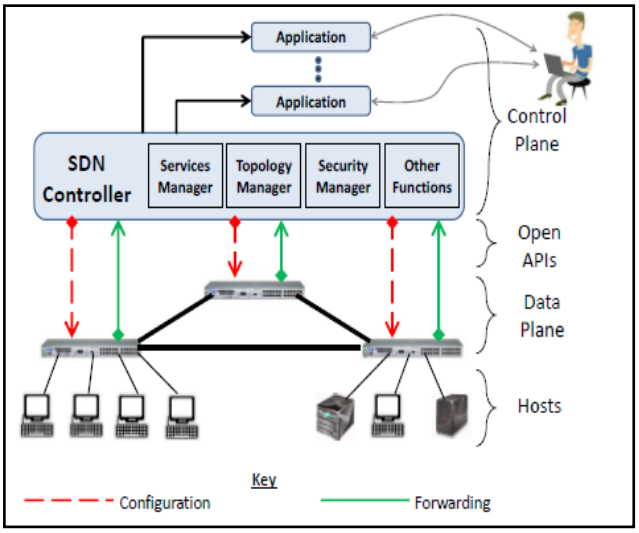
Abstract
Software Defined Networking (SDN) is an emerging technology that attracts significant attention from both industry and academia recently. Network security is a predominant topic both in academia and industry. Many methods and tools have been proposed but the attackers are still able to launch massive and effective attacks. The Software Defined Networking (SDN) paradigm introduces separation of data and control planes for flow-switched networks and enables different approaches to network security than those existing in present IP networks. We believe SDN provides new research opportunities to security, and it can greatly impact network security research in many different ways. However, till today, SDN has not been well recognized by the security community yet. Software Defined Networking (SDN) has recently emerged and promotes the programmability of the networks, which thus allows to enable in-network security functions. This includes firewalls, monitoring applications support through OpenFlow devices. Therefore, this paper reviews the related approaches which have been proposed by identifying their scope, their practicability, their advantages and their drawbacks. In this paper we analyse features of SDN in the context of security application. Additionally we point out some aspects of SDN networks that, if changed, could improve SDN network security capabilities.

What is The SDN

Software defined networking (SDN) is an approach to using open protocols, such as OpenFlow, to apply globally aware software control at the edges of the network to access network switches and routers that typically would use closed and proprietary firmware.

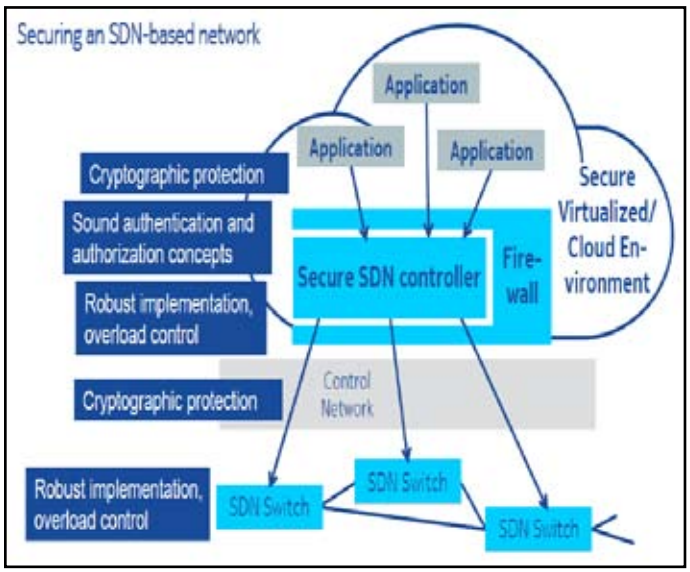
The Architecture of SDN

Computer networks typically consist of hosts interconnected by switches and routers providing data forwarding and routing functionality.The recently emergent Software Defined Networking (SDN) paradigm addresses this challenge by separating the packet forwarding functionality of the forwarding devices, i.e. the data plane from the control element, the control plane. This decoupling enables a radical new network architecture: switches in the network are reduced to basic packet forwarding devices containing flow tables populated with localized flow rules. These rules describe how incoming packets will be handled based on matching fields (such as packet header content, incoming port, etc.), and are managed by a remote 'controller' entity, which communicates securely with switches potentially using a standard and open interface, such as the OpenFlow protocol ..



Security Configuration Using SDN

Centralizing the Control Plane: The original vision for software defined network security management is spelt out by Casado et. al in SANE (Secure Architecture for Network Enterprise), a clean-slate security solution for enterprise networks. Enterprises today face a barrage of ever-evolving security threats and have little choice but to rely on a combination of security solutions that are complicated, distributed, and limited in scope. Security policies are typically implemented as complex, topology-dependent access control lists. Trust is distributed across multiple components, such as switches, DNS servers, authentication services and each of these individual components need to be protected in turn. If a network element is compromised, an attacker may be able to identify vulnerabilities and obtain sensitive information about the network itself, such as the topology, the location of critical servers, etc. Furthermore, security enforcement at higher layers may be undermined by unsecured access at the lower layers.



Behind the SDN paradigm that separates the control and data plane, SDN delivers four visible features to the networking field:

• **Central control and coordination** - the logically centralized

control model is a key part of the SDN architecture which mitigates the overhead from the traditional distributed mechanisms based on protocols. Although the centralized approach is often questioned for its scalability, it can deliver the state and policy changes more efficiently than the distributed methods in a managed domain. The coordination feature also makes it possible that when one of the controllers fails, other standby ones can take over the management tasks to avoid service breakage, which poses a great challenge for the distributed approach.

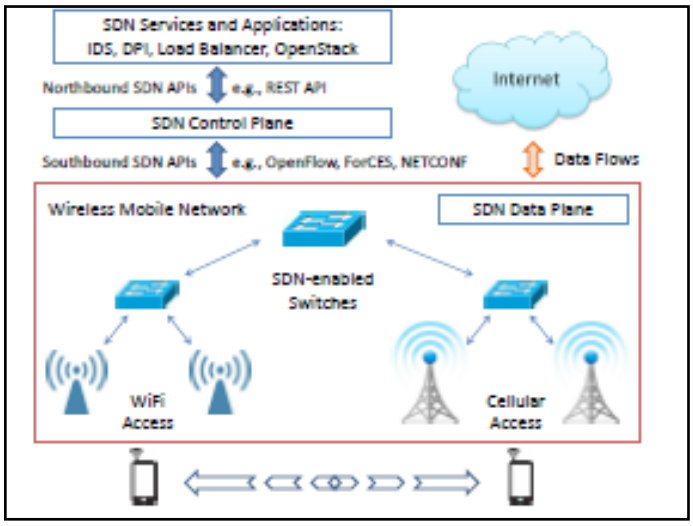
• **Programmability** - for both the control plane and the data plane, SDN makes implementation and deployment of the new functionality faster and easier, and hence speeding up the innovation at both hardware and software level. This agility can reduce the cost for service and network providers in terms of Operational Expenditure (OPEX) as the management can be powered by SDN applications in an automatic manner. By avoiding the unnecessary replacement of the underlying hardware through software update, it can also bring down the Capital Expenditure (CAPEX) and facilitate the adoption by the cloud providers.

• **Virtualized abstraction** - the layered design of SDN hides the complexity of hardware devices from the control plane and SDN applications. Through virtualized abstraction, SDN allows the managed network to be divided into virtual networks that share the same infrastructure but are governed by different policy and security requirements. Such flexibility greatly promotes the sharing, aggregation and management of available resources and enables dynamical reconfiguration and changes of policy.

• **Openness** - the open standards of SDN such as OpenFlow help build and develop open sourced communities that attract brain power and speed up the innovation. Such openness combined with programming APIs can promote the networking research by allowing researchers to experiment with novel ideas through fast prototyping and testing. It also benefits the interoperability with the legacy infrastructure and allows different operators and providers to collaborate through the SDN framework.

SDN for Wireless Mobile Networks

As wireless mobile networks are becoming the major channel to access Internet services, there is an urgent need to keep up with the pace of user growth and the scale of services. For instance, the recent demand of network capacity for mobile data traffic is far exceeding the supply of incumbent networks. At the same time, services also evolve in both variety and complexity. Since operators are limited by the commercial budget and the operation cost, it is extremely hard, if not impossible, to keep up with such speed while still cost-effectively upgrading the infrastructure, delivering service updates, and improving the end user experience under the existing infrastructure. As highlighted in Table I, we describe in this section the latest SDN solutions for wireless mobile networks that aim to address the challenges. The range of discussion covers the cellular and the WLAN environment, from the angle of core infrastructure and edge access.



Challenge And Requirement

The feature set in Table II provides several desired items. Based on the them, we identify the design challenges and outline the requirements.

• **Mobility and Roaming**: The mobility of users results in roaming between networks and potentially across different access technologies such as 4G and WiFi. This dynamic change adds complexity to the diagnose and detection of anomaly activities and as well as the security credential exchange

.

• **Monitoring Overhead**: The OpenFlow-based monitoring schemes suffer from limitation in terms of high overhead and incomplete sample information. FleXam provides a good example toward this challenge

.

• **Multi-Access and Multi-operator**: The operational environment consists of different technolgoies and operators leading to complex negotiation process, privacy concern,
and potential conflicting policy and QoS requirement that pose a challenge to the security enforcement.

• **Deployment**: Although SDN has openness in its nature, any solution deployed needs to face the challenge of backward compatibility and interoperability as operators need to maintain different generations of technologies (e.g., 3G and 4G) and intercommunicate with other providers. For a sound SDN design for security enhancement, we need to meet several requirement:

• **Interoperability**: Handling information exchange between different elements are crucial for SDN security design for wireless mobile environment. The recent trend of cellular offloading also makes this relevant since WiFi and Cellular management are used to be separate, especially the security part. Traditional distributed protocol is incapable for this due to the complexity and privacy issues.

• **Responsiveness**: Processing events in wireless mobile networks should be timely, either in reactive or proactive manner. Efficient triggering and local optimization are valuable.

• **Compatibility**: To maximize the value of openness, using a standard API is required, such as OpenFlow.

ISSN : 2347 - 8446 (Online)
ISSN : 2347 - 9817 (Print)

• **Adaptation**: Due to mobility and network condition changes, a design should be adaptive by monitoring and efficiently detecting events both in the network and from user activities.

• **Simplicity**: To promote deployability and encourage contributions from the community, a proposal should avoid complex extension on hardware-based data plane and OpenFlow protocol extension.

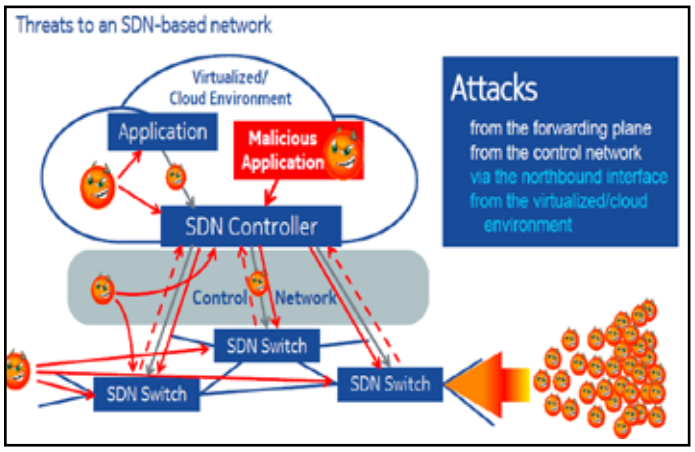
Securing The Data Plane

- MAC Address modification may be reported.
- The controller should be able to learn the tunnel IDs associated with logical ports.
- The controller should periodically collect the statistical information of ports.
- State transition of all SDN components should be logged. All logged information should be protected.
- Design flow-control mechanism to assure reliable updates and communications
- between controllers and switches (more research needed)
- Enforce message validation and integrity to avoid unintended consequences
- of misconfiguration of instantiation of corrupt table entries
- Implement a PKI CA to manage trust, authenticity, revocation and repudiation
- Ensure authenticity of communications endpoints within the OF SDN fabric (802.1x)
- Employ policy conflict resolution mechanisms at the controller.
- Secure switch storage of controller connection information.

Security Threats and Possible Attacks in SDNS

The configuration data alteration threat represents the destruction or alteration of configuration data that is required by SDN to perform different functions. Configuration data can be removed or modified from SDN platform. The threat can be mitigated by ensuring data integrity in SDN middleware. The threat can target the control layer, control-data interface, and data layer and can affect the resource and application management. The configuration data extraction threat is an eavesdropping threat, where an attacker gathers configuration data that can be used in subsequent attacks. The configuration data extraction requires confidentiality protection and ensuring data integrity functionality in SDN. The threat targets the control layer, control-data interface, and data layer. An unauthorized access to SDN services threat identifies a security breach, where an authorized SDN entity can access services of SDN for which the entity does not have the proper access level. The threat can be mitigated by deploying secure administration module and ensuring the system integrity. The security requirement for mitigating the threat is identification verification. The threat can affect all functionalities and can target all layers and interfaces of SDN.

Protecting flow paradigm of the SDN is grounded on flow based forwarding and can certainly ensure the end-to-end communication security.



The flow paradigm is acting as the soul of SDN and it must be protected. A successful injection of bogus flow may lead to the entire network disaster. The flow abstraction shown by the controller may easily lead in harvesting the intelligence of the connected resources, which can be used in further attacks and exploitations. An up to date access control mechanism should be deployed in the network. Moreover, flows should be encrypted to avoid injecting malicious flows. Proper authentication and authorizations should be implemented to avoid side channel attacks.

Conclusions

Software Defined Networking has much to offer security applications. However, current approaches to building SDN based security applications are not practical because of performance limitations and deployment hurdles. Defined Networking has much to offer security applications. However, current approaches to building SDN based security applications are not practical because of performance limitations and deployment hurdles. The OFX framework provides a better approach that overcomes these challenges. It allows security applications to improve performance by extending switches with custom functionality and works within existing OpenFlow infrastructures.

References

[1] *S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: measurements & analysis,"*

[2] *A. D. Ferguson, A. Guha, C. Liang, R. Fonseca, and S. Krishnamurthi, "Participatory networking: An api for application control of sdns," in ACM SIGCOMM Computer Communication Review, vol. 43, no. 4. ACM, 2013, pp. 327–338.*

[3] *V. Heorhiadi, M. K. Reiter, V. Sekar. New Opportunities for Load Balancing in Network-wide Intrusion Detection Systems. In Proceedings of ACM CoNEXT 2012.*

[4] *J. H. Jafarian, E. Al-Shaer, Q. Duan. OpenFlow Random Host Mutation: Transparent Moving Target Defense using Software Defined Networking. In Proceedings of ACM HotSDN 2012.*

[5] *Y. Wang, Y. Zhang, V. Singh, C. Lumezanu, G. Jiang. NetFuse: Shortcircuiting Traffic Surges in the Cloud. In Proceedings of IEEE ICC 2013.*

[6] *S. Shin, G. Gu. CloudWatcher: Network security monitoring using OpenFlow in dynamic cloud networks (or: How to provide security monitoring as a service in clouds?). In Proceedings of IEEE ICNP NPSec Workshop 2013.*

[7] *Jonathan Warren. Bitmessage: A Peer-to-peer Message*

Authentication and Delivery System. White Paper (27 November 2012),

[8] *N. Feamster, M. Balazinska, G. Harfst, H. Balakrishnan, and D. R. Karger. Infranet: Circumventing web censorship and surveillance. In USENIX Security Symposium, pages 247–262, 2002.*