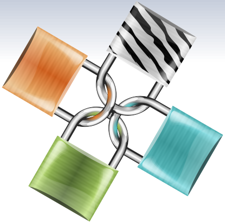


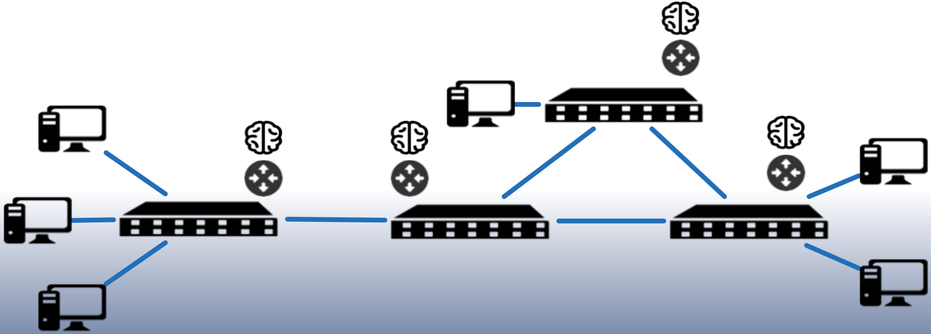
Securing the SDN Northbound Interface

With the aid of Anomaly Detection

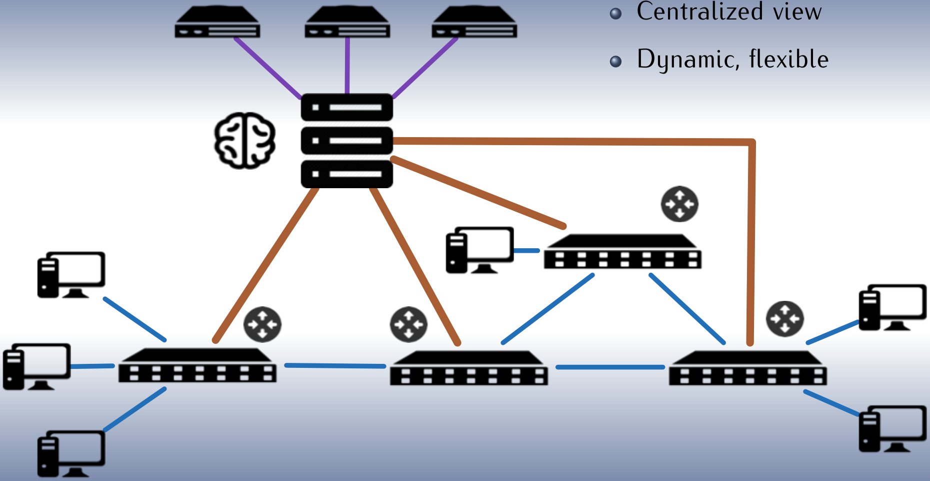
Jan J. Laan

July 2, 2015

Traditional network



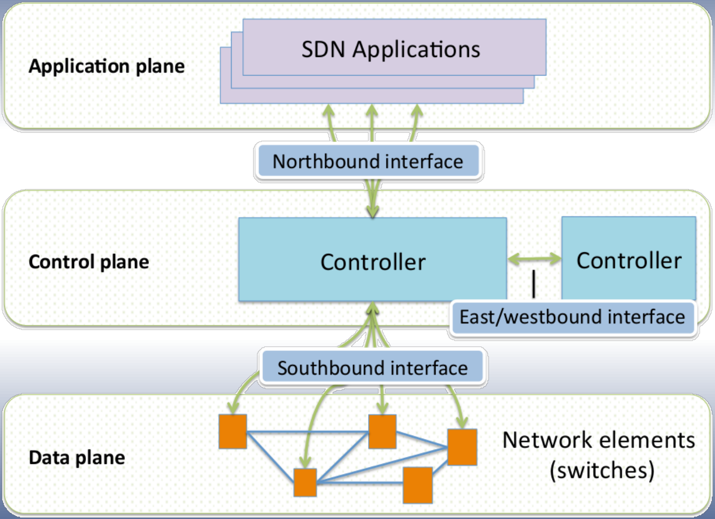
SDN network



Advantages

- Centralized view
- Dynamic, flexible

SDN overview



Application plane

SDN Applications

Northbound interface

Control plane

Controller

Controller

East/westbound interface

Southbound interface

Data plane

Network elements (switches)

Research question

Main question

How to perform a security assessment of the northbound interface of a SDN network?

Supporting questions

- What are the main threats, and associated security requirements, to the SDN northbound interface?
- What is the best approach to assess the security of a northbound interface?
- How secure are the northbound interfaces of current popular SDN controllers?
- How can best practices with regard to security be improved?

Related work

OperationCheckpoint [1]

Northbound Access control for the Floodlight controller

SEFloodlight [2]

Conflict resolution, authentication for the Floodlight controller NB API.

Rosemary [3]

A controller built with security by design, especially for the northbound interface.

Testbed

5 popular and/or interesting controllers for testing.

Project Floodlight
onos
Open Network Operating System
OPEN DAYLIGHT
open MUL

1: HTTPS support

Goal: Secure communication in the northbound interface

Check for supported HTTPS versions

[1] Web interface stops working

[2] SSL3 enabled

1: HTTPS support

Goal: Secure communication in the northbound interface
Check for supported HTTPS versions

Floodlight	Onos	OpenDaylight	Ryu	Open Mul
Yes	Yes	Yes[1]	No	Partial[2]

[1] Web interface stops working
[2] SSL3 enabled

Current status

2: Authentication

Goal: Only allow access to authorized users/applications

2: Authentication

Goal: Only allow access to authorized users/applications

Floodlight	Onos	OpenDaylight	Ryu	Open Mul
Yes	Yes	Yes	No	No

Floodlight, Onos and OpenDaylight: Client certificates
OpenDaylight: HTTP Basic

3: Authorization

Goal: A user/application can only access the parts of the API he needs.

3: Authorization

Goal: A user/application can only access the parts of the API he needs.

Floodlight	Onos	OpenDaylight	Ryu	Open Mul
No	No	No	No	No

Research project for Floodlight with access control.

4: Logging

Goal: non-repudiation, there is a trail of access to the northbound interface.

4: Logging

Goal: non-repudiation, there is a trail of access to the northbound interface.

Floodlight	Onos	OpenDaylight	Ryu	Open Mul
Yes	Yes	Yes	No	No

5: Documentation

Goal: Ease of configuration for security features

5: Documentation

Goal: Ease of configuration for security features

Floodlight	Onos	OpenDaylight	Ryu	Open Mul
Yes	No	No	No	No

Results summary

	Floodlight	Onos	OpenDaylight	Ryu	Open Mul
HTTPS	Yes	Yes	Yes	No	Partial
Authentication	Yes	Yes	Yes	No	No
Authorization	No	No	No	No	No
Logging	Yes	Yes	Yes	No	No
Documentation	Yes	No	No	No	No

Insecure by default. Almost all security features are turned off initially.

Malicious applications

A scenario:

1. Application has access through the northbound interface
2. Application gets hacked
3. Hacker abuses access rights to disrupt the network
4. Security measures mentioned before will not prevent this

Malicious applications

A scenario:

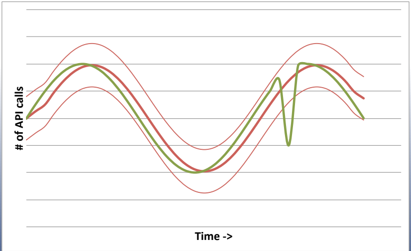
1. Application has access through the northbound interface
2. Application gets hacked
3. Hacker abuses access rights to disrupt the network
4. Security measures mentioned before will not prevent this

Possible solution: **Anomaly detection**

Premise: When an application becomes malicious, its behaviour changes.

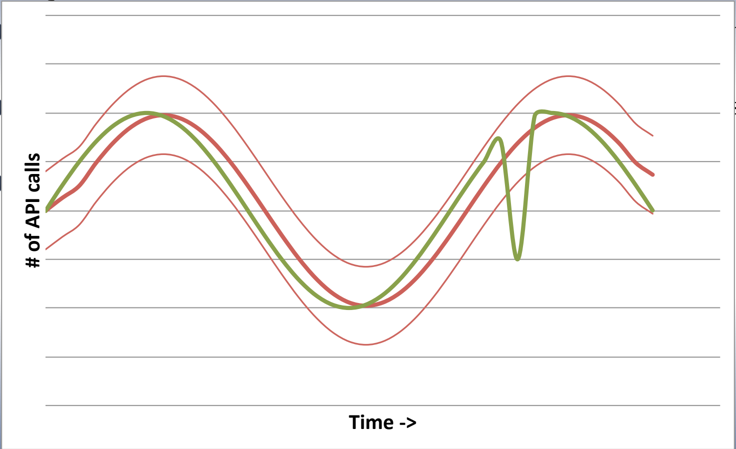
Statistical Anomaly Detection

1. Log all access to northbound interface
2. Divide data into ”historical” (training) data and ”current” (testing) data.
3. Compare weighted chances per API call per application for these data sets.
4. Calculate an anomaly score.

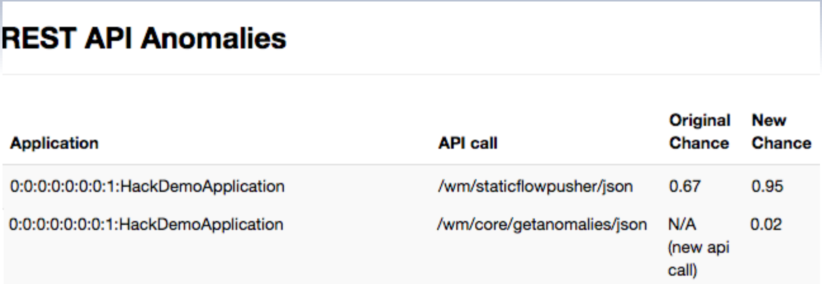


Statistical Anomaly Detection

1. Log all access to northbound interface
2. ng)
3. se
4.



Floodlight Proof of Concept



REST API Anomalies

Application	API call	Original Chance	New Chance
0:0:0:0:0:0:0:1:HackDemoApplication	/wm/staticflowpusher/json	0.67	0.95
0:0:0:0:0:0:0:1:HackDemoApplication	/wm/core/getanomalies/json	N/A (new api call)	0.02

Performance impact: 7% (1.1ms extra latency)

Needs further research for validation.

Limitations

- Only works well for predictable applications.
- Can be ”trained” to accept malicious behaviour.
- Requires parameter tuning.

Conclusion

SDN northbound interface security is poor at this time.

Adding access control and turning on other tested features will help.

Insecure by default, lack of security features.

Anomaly detection: interesting addition, needs further research.

Future work

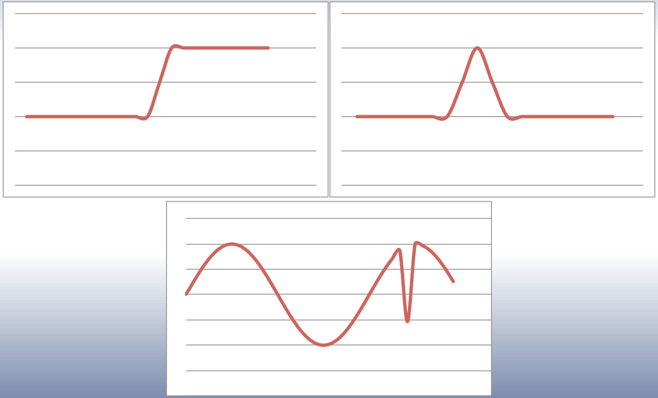
- Implement authorization on controllers.
- In-depth analysis of a single controller.
- Validate detection rate of statistical anomaly detection
- Explore other means of anomaly detection (machine learning, data mining)

References

- S. Scott-Hayward, C. Kane, and S. Sezer, “Operationcheckpoint: SDN application control,” in Network Protocols (ICNP), 2014 IEEE 22nd International Conference on, 10 2014, pp. 618–623.
- P. Porras, S. Cheung, M. Fong, K. Skinner, and V. Yegneswaran, “Securing the software-defined network control layer,” in Proceedings of the 2015 Network and Distributed System Security Symposium (NDSS), San Diego, California, 2015.
- S. Shin, Y. Song, T. Lee, S. Lee, J. Chung, P. Porras, V. Yegneswaran, J. Noh, and B. B. Kang, “Rosemary: A robust, secure, and high-performance network operating system,” in Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security, ser. CCS ’14, New York, NY, USA, 2014, pp. 78–89.

Anomaly types

The red line depicts the amount of API calls over time to an API function. Three types of anomalous traffic are shown.



Security assessment (STRIDE)

Spoofing

- (Lack of) user authentication
- Divert NB network traffic. (f.e. ARP spoofing)

Tampering

- Capture and alter network traffic (MitM)
- take over (hack) SDN application

Repudiation

- Log API access

Security assessment (STRIDE) cont.

Information disclosure

- Listen in on network traffic

Denial of Service

- Send many requests to the NBI.
- Request resource-intensive tasks from controller.

Elevation of Privilege

- Access unauthorized parts of the API