A Different Kind of SDN: Self-Driving Networks

Walter Willinger
NIKSUN, Inc., Princeton, NJ
wwillinger@niksun.com

December 7, 2017

- A Concrete Use Case
- Queries as First-class Citizens
- SONATA – A Query-driven Network Telemetry System

A Concrete Use Case

- DoD's Joint Information Environment (JIE)

- http://www.disa.mil/~/media/Files/DISA/News/Conference/2016/AFCEA-Symposium/1-ColJackson_JRSS.pdf
- https://www.govtechworks.com/jrss-adds-security-features-pushes-ahead-in-u-s-and-europe/#gs.6XAGbHs
- http://www.prnewswire.com/news-releases/igov-and-niksun-partner-to-provide-critical-technology-for-disa-jrss-300238339.html

DoD’s Joint Information Environment (JIE)

- Sweeping, multi-year effort to re-design DoD’s disjointed conglomeration of networks into a single, secure, Joint Information Environment (JIE)
- A single joint enterprise IT platform that can be leveraged for all DoD missions
- Defense Information System Agency (DISA) is the technical and implementation lead for JIE

- (Non-classified) reference document
  - “Enabling the Joint Information Environment (JIE)”, 05/0514
  - http://www.disa.mil/~/media/Files/DISA/About/JIE101_000.pdf

Some JIE Specifics

- Design goal
  - **Today**: There are some 1000+ loosely controlled entry points into DoD's conglomeration of networks
  - **Future**: All network traffic into and out from DoD networks will flow through 48 gateways or entry points called Joint Regional Security Stacks (JRSS)

- Traffic rates
  - Some JRSSs are designed to handle 10Gbps
  - Most JRSSs are required to handle 40 or 50Gbps

More JIE Specifics

- JIE is a global network
  - 22 JRSSs are located in the US
  - 6 in EU
  - 5 in SWA
  - The rest in the Pacific region

- JIE supports unclassified and classified traffic
  - 23 JRSSs are used for unclassified network traffic
  - 25 are dedicated to handle classified traffic

More about JRSS

- The 48 JRSSs are intended to become DoD’s bulwark against hackers, viruses and malware attacks

- Originally required stack capabilities (JRSS 1.0, FY 2014)
  - Load balanced FWs
  - IPSs, IDSs, alerts/logs, etc.
  - Traffic tap for passive sensing

- New requirement (JRSS 1.5, FY 2015)
  - Full (and loss-less) packet capture for deep packet analysis

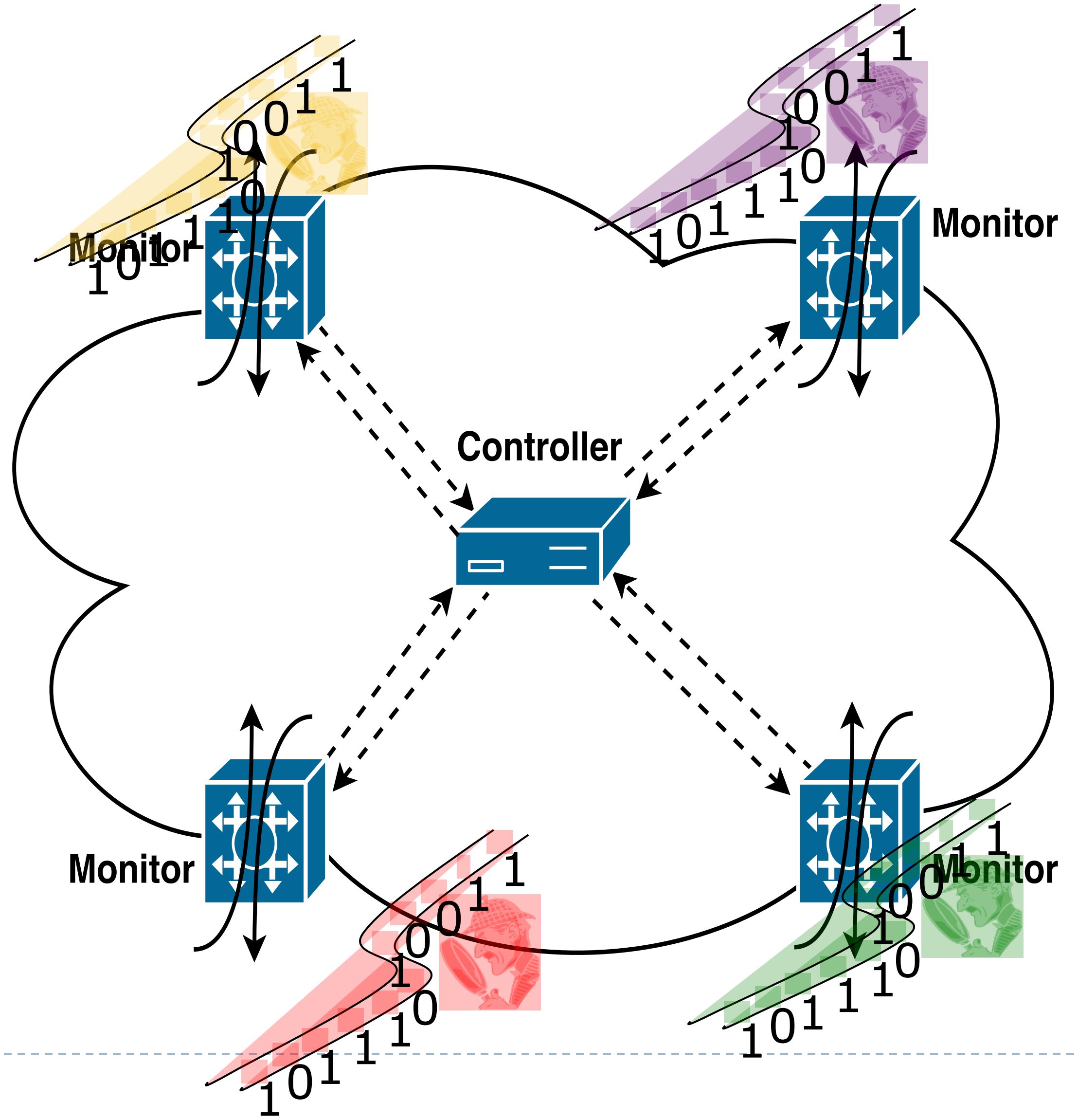
JIE-motivated Research Problems

- Given: "All packets, all the time" is a solved problem
  - NIKSUN's Industry-leading technology enables loss-less full packet capture at scale (up to 100 Gbps and beyond)
  - Capturing every packet that enters or leaves your network is feasible
  - This raw data and various associated on-the-fly generated metadata can be stored and indexed in a data warehouse

A New Type of "Big (Internet) Data"

- Implications of an "all packets, all the time" solution
  - The resulting "big data" is of the streaming type (e.g., 50Gbps)!
  - The resulting "big data" is in addition distributed (e.g., 48 locations)!
- Requirements for resulting distributed streaming data
  - No moving of "big streaming data" to a central location!
  - No multiple passes over "big streaming data" at remote locations!
  - Remote locations are "beefy" (i.e., resource-rich)!
  - Assume "Command & Control"-like communication structure!

Distributed Streaming Data is the Future!



Distributed Streaming Data Algorithms

- Questions are formulated as queries that are launched from a central node

- Queries are designed to provide specific results that concern the union of all distributed streaming data without having access to all this data in one place

- Launching a query typically means fetching pertinent information maintained by each of the different remote nodes about certain aspects of their data and then use the obtained information to provide the answer to the original query (at the central node)

“Big Data” for the Good of Networking

- Security-related network management applications
  - (Close-to) real-time detection of the onsets of different types of network attacks, followed by swift and effective (e.g., minimal collateral damage) mitigation strategies
- Performance-related network management applications
  - (Close-to) real-time detection of user-experienced service degradations as they happen, followed by instructions for the network to perform corrective steps in a (close-to) real-time and purposeful manner
- A key step towards “self-driving networks” …

Networking for the Good of "Big Data"

- Systems support of distributed streaming data analytics
  - Streaming data processing platform at remote locations for efficiently processing queries over local data streams
  - Infrastructure to enable the communication between each remote location and the controller to answer queries concerning the union of all distributed streaming data

- Systems support for driving real-time control decisions
  - Coupling of distributed streaming data analytics platform with programmable data plane capabilities
  - Programmable switches (e.g., SDN) for installing forwarding table rules that affect how traffic is forwarded

Research Opportunities - Algorithms

- Cyber Security – the new "killer application" for distributed streaming data algorithms?
  - Development of new distributed streaming data algorithms
  - Approximate queries/answers are often "good enough"
  - Trading off communication cost vs accuray of answers
  - Towards automated detection/mitigation of known types of attacks in (close-to) real-time
  - Support for exploratory data mining (for network forensics) and discovery (for learning about new types of attacks) through off-line "back-in-time" analysis
  - Support for visualization-oriented queries

Research Opportunities – Databases

- Design of query processing engine in support of
  - Continuous and one-time queries
  - Pre-defined and ad-hoc queries
  - Exact and approximate queries
- Design of streaming data processing platform that supports
  - Dynamic query modification
  - Running a large number of simultaneous queries
  - “Time travel” or “back-in-time” analysis

Research Opportunities - Networking

- Programmable data plane
  - PISA – protocol-independent switch architecture (Barefoot Networks)
  - P4 – language for programming protocol-independent packet processors
  - SDN – software-defined networking
  - …

Back to JIE Use Case …

- Envisioned new capabilities (JRSS 2.0, target date 2019)
  - Greater fidelity of cyber vulnerability screening and filtering
  - Data/Applications defense in depth
  - Detonation chamber or dynamic execution environment
  - Advanced and fine-grained inspection of all of DoD's traffic
  - WAN traffic optimization for data/application support
  - Towards fully autonomic defensive systems
  - Real-time visualization of attack detection and mitigation "in action"
  - Etc.

Queries as First-Class Citizens

- *Joint with A. Gupta, R. Harrison, A. Pawar, R. Birkner, M. Canini, N. Feamster, and J. Rexford*

Cyber Security Grand Challenge

- DARPA’s effort
  - Cyber Grand Challenge (CGC) 2013-2016
  - A competition to create automatic defensive systems capable of reasoning about flaws, formulating patches and deploying them on a network in real time

- Our (more modest) goal
  - Designing systems for the detection **and** mitigation of known and unknown nefarious activities in tandem and in almost real-time
  - Well-known challenges
    - Detection is typically based on weak signals
    - False alerts (F/P’s and F/N’s) should be avoided/eliminated
    - Collateral damage should be quantified/controlled/eradicated

Artificial Intelligence (AI) and Cyber Security

Darktrace is
security. Cr
Cambridg

…ing machine learning company for cyber
…s from the University of
…une System uses AI
…erprise

Cylance prevents cyberattacks
CylancePROTECT leverag…
and machine learning to
attack prevention th…
the cloud,

…mic sci…
…ylance

deepinstinct is the first com…
cybersecurity to prov…
endpoints and

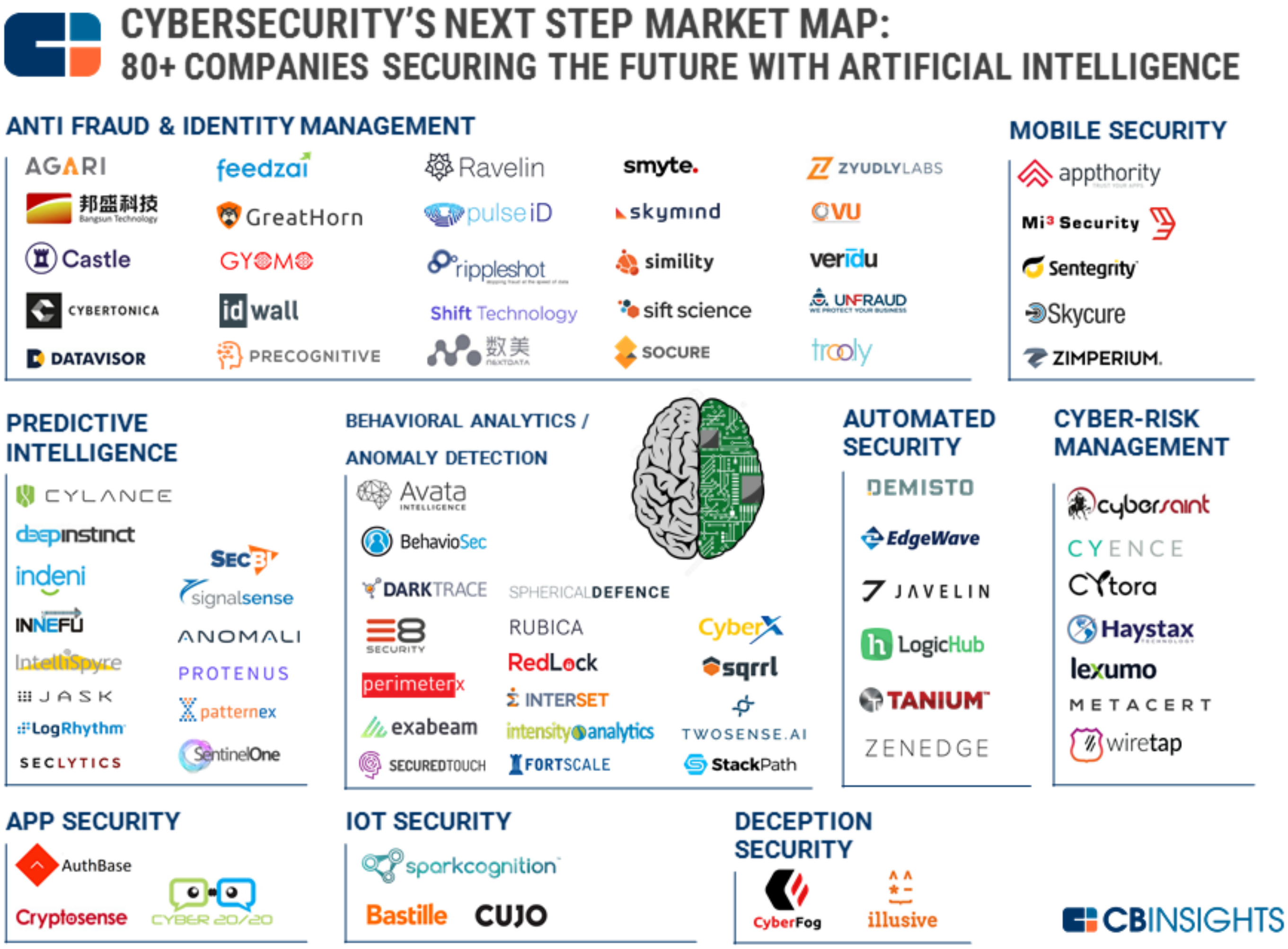
…on for

SparkCognition is a global leader in cognitive computing
analytics. A highly-awarded company recognized for cutting-
technology, SparkCognition develops AI-Powered cyber-
…s for the safety, security, and reliability of IT,
…kcognition.com

IBM's new Cognitive SOC platform embeds Watson for Cyber Security's unique ability to understand, reason and learn about security topics and threats. The Cognitive SOC connects obscure data points humans can't possibly identify on their own, enabling enterprises to quickly and accurately respond to threats across networks, endpoints, users and cloud.

www.ibm.com

The Allure of AI for Cyber Security



CYBERSECURITY'S NEXT STEP MARKET MAP:
80+ COMPANIES SECURING THE FUTURE WITH ARTIFICIAL INTELLIGENCE

ANTI FRAUD & IDENTITY MANAGEMENT

AGARI, 邦盛科技 Bangsun Technology, Castle, CYBERTONICA, DATAVISOR, feedzai, GreatHorn, GYOMO, idwall, PRECOGNITIVE, Ravelin, pulseiD, rippleshot, Shift Technology, 数美 NEXTDATA, smyte., skymind, simility, sift science, SOCURE, ZYUDLYLABS, CVU, veridu, UNFRAUD, trooly

MOBILE SECURITY

appthority, Mi3 Security, Sentegrity, Skycure, ZIMPERIUM.

PREDICTIVE INTELLIGENCE

CYLANCE, deepinstinct, indeni, INNEFU, IntelliSpyre, JASK, LogRhythm, SECLYTICS, SECBI, signalsense, ANOMALI, PROTENUS, patternex, SentinelOne

BEHAVIORAL ANALYTICS / ANOMALY DETECTION

Avata INTELLIGENCE, BehavioSec, DARKTRACE, E8 SECURITY, perimeterx, exabeam, SECUREDTOUCH, SPHERICALDEFENCE, RUBICA, RedLock, INTERSET, intensity analytics, FORTSCALE, CyberX, sqrrl, TWOSENSE.AI, StackPath

AUTOMATED SECURITY

DEMISTO, EdgeWave, JAVELIN, LogicHub, TANIUM, ZENEDGE

CYBER-RISK MANAGEMENT

cybersaint, CYENCE, CYtora, Haystax, lexumo, METACERT, wiretap

APP SECURITY

AuthBase, Cryptosense, CYBER 20/20

IOT SECURITY

sparkcognition, Bastille, CUJO

DECEPTION SECURITY

CyberFog, illusive

CBINSIGHTS

The Reality of AI for Cyber Security …

- AI is truly amazing for tasks such as …
  - Photo tagging (FB), movie recommendations (Netflix), …
- But cyber security is different …
  - Requires unsupervised learning, based on weak signals
  - Has stringent constraints (real-time, false alerts)
  - Often with limited opportunities for learning
- Cyber security in practice
  - Widely-acknowledged "alert fatigue"
  - Exorbitant dwell times (still about 200 days in 2016; Equifax!)

Our Take on the Grand Challenge Problem

- What is needed (in words)?
  - A new purposefully-design system/platform that facilitates easy user interactions (i.e., writing and running queries) with the type of data that can be collected in today's network (i.e., (distributed) streaming data) to find "the needles in the haystack" (i.e., the data that satisfies the expressed queries) in close-to real-time and with high confidence.

- What is needed (in terms of technologies)?
  - "Smart" compiler and run-time systems that allow users to write queries in a high-level language without having to worry about how and where they get executed.

“Thinking architecturally” about Queries

- #1: View queries as first-class citizens
- #2: Develop a simple (but expressive) query language
- #3: Design for scalability
  - Expect to run 1000’s of concurrent queries
  - Take high data rates (Gbps and beyond) as given
- #4: Optimize everything and everywhere
- #5: Design for change
  - Programmable data plane (e.g., OF, P4)
  - Stream processing engines/platforms (e.g., Storm, Spark)

Illustration: DNS Amplification Attack (1)

- Traditional approach ("queries as 2nd class citizens")
  - Focus is on detection, no concern for mitigation
  - A simple case of a volume-based anomaly detection query
    - For each destIP, count the number of unique srcIPs sending DNS response messages
    - Output all destIPs for which this count is higher than threshold k1

- Key observations ("querying for the sake of querying")
  - Easy to write …
  - Operates on packet header only
  - Likely to produce a large number of false positives
  - Almost certain to result in excessive collateral damage

Illustration: DNS Amplification Attack (2)

- Our approach ("queries as 1st class citizens")
  - Focus is **end-to-end**, on detection & mitigation
  - Stage 1 (same as before)
    - For each destIP, count number of unique srcIPs sending DNS response
  - Stage 2 (make use of additional information)
    - For those destIPs for which the count in Stage 1 is higher than threshold k1, count the number of responses with resource record type *RRSIG* in DNS response's header field (part of UDP packet's payload)
    - Output all destIPs for which the count in Stage 2 exceeds threshold k2

- Key observations ("querying with a purpose")
  - This query is the join of two simple sub-queries
  - Using additional information in Stage 2 (e.g., in payload of UDP packets) allows for controlling false alerts and in turn collateral damage

Design Principle #1:
Queries as First-class Citizens

- Consider a query's task **end-to-end**
  - From detection to mitigation
- Control false alerts and implied collateral damage
  - Selectively incorporate additional evidence/relevant information
  - Additional evidence is query-specific and can range from fine- to coarse-grained (e.g., packet to flows to sessions/applications), check for temporal patterns ("stepping stones"), or tap into other meta-data
- Highlights key role of (flexible) join operator
  - For structuring queries in terms of different simple sub-queries
  - For expressing increasingly complex queries

A Side Remark: “Intrusion Kill Chain” (I)

- A rough guide for breaking down attacks
- An attempt to base security decisions and measurements on a keen understanding of the adversary

- Benefits
  - One mitigation anywhere along the chain breaks the chain!
  - Objective is to raise the adversaries’ costs to achieve their goals (i.e., how many elements in the chain need to be changed to succeed with the next intrusion?)

A Side Remark: “Intrusion Kill Chain” (II)

- Reconnaissance
- Weaponization
- Delivery
- Exploitation
- Installation
- Command & control
- Actions on objectives

- http://www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM- White-Paper-Intel-Driven-Defense.Pdf

Design Principle #2:
Need for a high-level Query Language

- Writing queries should be easy, because with ease-of-use comes broad adoption by users

- Leverage common paradigms whenever possible
  - (Extensible) packet-as-tuple abstraction
  - Popular MapReduce-like dataflow programming model
  - Join operator as key building block for writing complex queries
- Queries as compositions of dataflow operators over packet tuples
  - Amenable to a wide range of cyber security-specific tasks
  - But also: network performance, traffic engineering

Design Principle #3:
Querying at Scale

- With broad adoption by operators/cyber security analysts comes the need to be able to query at scale …
- Design for running 1000's of concurrent queries …
  - Continuous and periodic queries
  - Ad-hoc and triggered queries
  - Dynamic queries
- … and returning the answers in close-to real-time!
  - For high-velocity streaming data (from Gbps to Tbps)
  - With heterogeneous data (packets, flows, applications, metadata)
  - In complex environments (single switch vs distributed settings)

Design Principle #4:
Query-related Optimizations

- For querying at scale, compute resources need to be allocated and used very efficiently and effectively …

- Build "smart" run-time systems (i.e., opportunities to optimize)
  - What data structure (synopsis) for what types of queries?
  - How to iteratively "zoom-in" on traffic of interest?
  - Where to execute which dataflow operations of which (sub)query?
  - How to benefit from overlapping queries?
- Understand (inevitable) trade-offs
  - Approximate answers vs speed vs resource consumption
  - How and when to learn about and adapt to changing workloads?

Design Principle #5:
Queries amid Changing Technologies

- Ensure that the system can evolve so as to embrace/exploit the latest technological advances

- Programmable data plane targets
  - OF-based: filter and sample operations
  - PISA-based: filter, sample, map, reduce, join operations
- Stream processor targets
  - Apache Spark: micro-batching, includes MLlib
  - Apache Storm: over 1 million tuples per sec (benchmarking)
  - Apache Flink: processes data streams as true streams

SONATA: A First-Principles Approach to Query-Driven Network Telemetry

- Design Principles #1 and #2

  → Uniform programming abstraction ensures ease-of-use

- Design Principles #3 and #4

  → Novel optimization approaches ensure scalability

- Design Principle #5

  → Modularity allows compilation to different targets

SONATA: Query-Driven Network Telemetry

- Application Interface
  - Express queries a dataflow operations over packet tuples
  - Write queries w/o worrying about where/how they get executed

- Compiler and run-time systems
  - Iteratively refine and partition each input query and identify and exploit overlap among multiple queries
  - Compile "best" query plan to target-specific configurations

- Big data environment
  - (Distributed) streaming data

Summary

- SONATA enables **expressive** and **scalable** network telemetry using
  - Declarative Query Interface
  - Query Partitioning
  - Iterative Refinement

- Check out/Contribute to Sonata Project
  - 10+ active members and growing
  - GitHub: github.com/Sonata-Princeton/SONATA-DEV

sonata.cs.princeton.edu

Upcoming Workshop

- SIGCOMM 2018 Workshop of "Self-Driving Networks"
  - Budapest (Hungary), August 19-24, 2018
  - Organizers: N. Feamster, J. Rexford, W. Willinger
  - Submission deadline: End of March of 2018