Cloud Security Monitoring on a Dime Store Budget

Wes Lambert

DEFCON Blue Team Village 2020

Agenda

- NSM
  - Core concepts
  - Data types
  - Challenges
  - Security Onion
- Native Facilities for NSM in the Cloud
- Automating deployment of NSM in the cloud

NSM – Network Security Monitoring

...collection, analysis, and escalation of indications and warnings to detect and respond to intrusions…

https://taosecurity.blogspot.com/2014/09/a-brief-history-of-network-security.html

NSM – Cycle

- Collection
- Detection
- Analysis

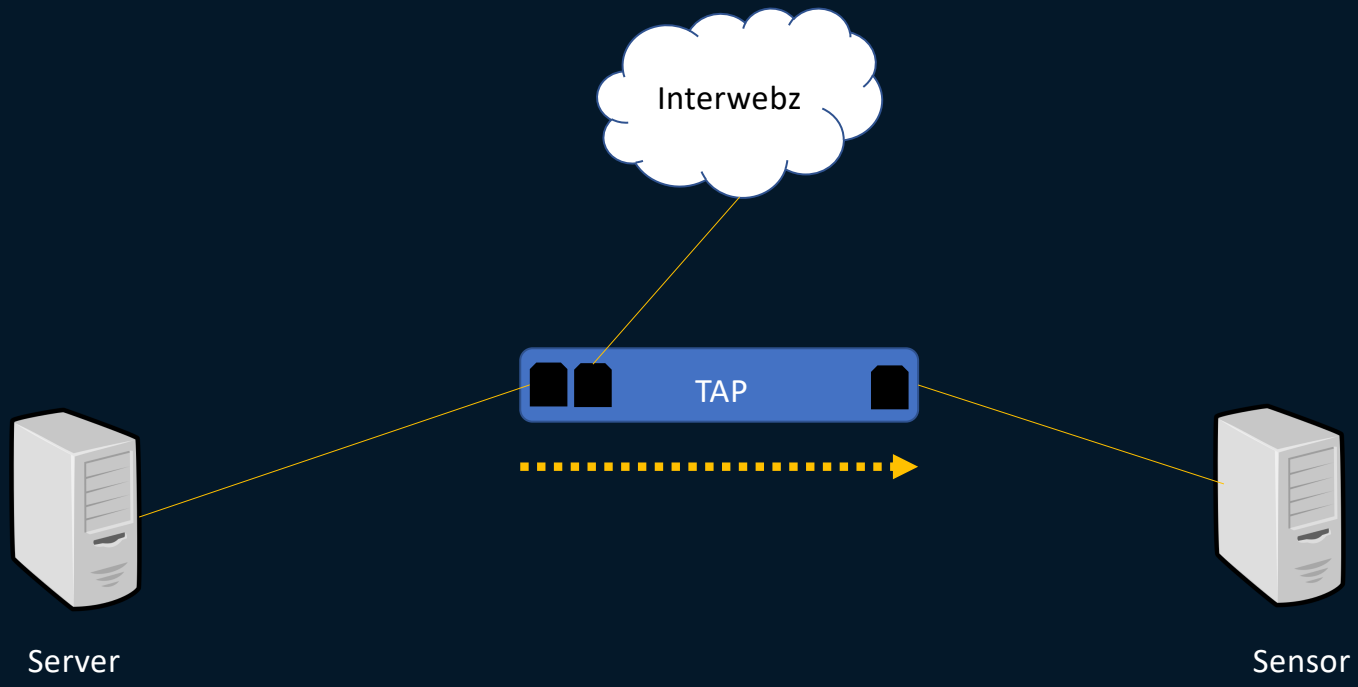
Ref: Applied NSM – Chris Sanders, Jason Smith

NSM - Collection

- Most important phase of the NSM cycle
  - Need data to facilitate detection and analysis!
- Can often be the most labor intensive phase, due to coordination amongst leadership, and various technical teams
- Can include tasks such as:
  - Defining where the largest amount of risk resides within an organization
  - Identifying threats to organizational goals
  - Identifying/refining relevant data sources
  - Configuring capture and retention of data (hardware/software/storage)

Ref: Applied NSM – Chris Sanders, Jason Smith

NSM – Collection - Typical Infrastructure



NSM - Detection

- Process by which collected data is examined and alerts are generated based upon observed events and data that are unexpected.
  - Signatures
  - Statistical anomalies
- Most often completed by software
  - NIDS
    - Snort/Suricata/Zeek
  - HIDS
    - Wazuh/OSSEC
  - SIEM
    - Sigma
    - Built-in correlation

NSM – Analysis

- Final stage of NSM lifecycle
- Where humans interpret and investigate alert data
- Usually involves gathering data from additional sources
  - HIDS
  - NIDS
  - Windows Event Logs/Sysmon
  - HTTP/DNS transactions
- May ultimately include malware analysis

NSM – What it is

- Obtaining a copy of each packet traversing the network and forwarding to a sensor for collection and analysis
- Mechanism to provide rich, contextual data about interactions between hosts within a monitored computer network or segment
- Relatively inexpensive to implement, compared to other methods of security monitoring

NSM – What it isn't

- NSM is not intended to prevent intrusions
  - Prevention eventually fails
  - s/prevention/detection/
- NSM is not EDR, although can be integrated alongside for greater visibility
- NSM is not "continuous monitoring"
  - NSM == threat centric
  - CM == vulnerability-centric
- NSM is not just having a magic box to find all badness

NSM – Data Types

- Full Content Data
  - Entire conversation between two hosts
    - PCAP
- Extracted Content
  - Files extracted from network stream
    - EXEs
    - PDFs
    - RARs

NSM – Data Types (2)

- Session Data
  - Details about a particular network session/connection seen between two hosts
    - Number of bytes
    - Duration
    - Connection state
    - Protocol/service type

NSM – Data Types (3)

- Transaction Data
  - Data about a particular type of network transaction
    - HTTP
      - URI/User Agent/Method/etc
    - DNS
      - Query/Answers/Response Code
- Alert Data
  - Data generated by an IDS, alerting the user to a particular content match
    - Ex. Notification for detection of Trickbot Trojan

NSM - Challenges

- NAT
  - Can obscure source/destination addresses
- Encryption
  - Can hamper the ability to see contents of traffic
- Throughput
  - Hardware/performance implications
- Privacy
  - Legal implications

NSM – In the cloud

- NSM in the cloud is still a toddler
- Some providers provide native mechanisms, while others do not
  - In the event that a native mechanism is not available, workarounds do exist, but can be difficult to maintain and scale
- NSM can be complemented by other sources of telemetry, for a more wholistic approach

NSM – Cloud Challenges

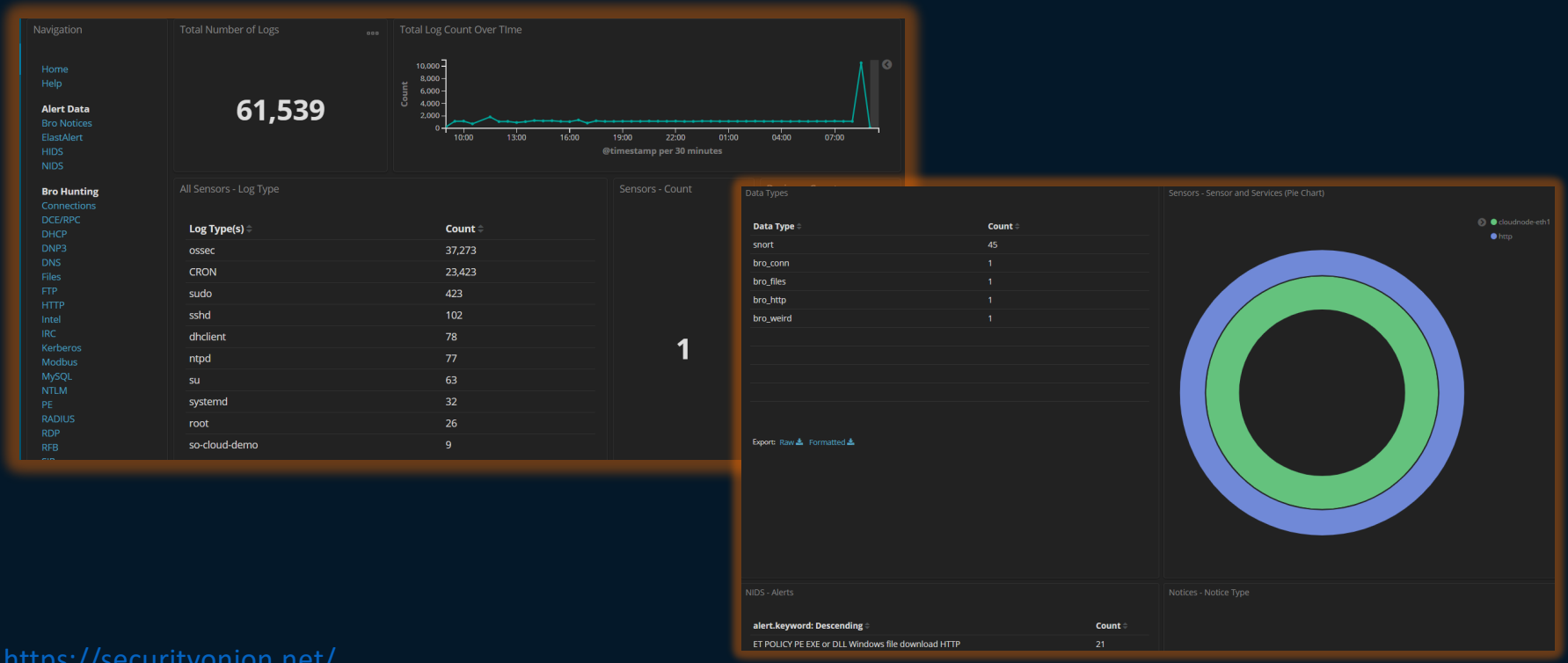
- VXLAN encapsulation
  - We can record it, but can our applications handle it?
- Segmentation
  - Are we able to get the data we need to our sensors?
- Ephemerality
  - Services in the cloud frequently spin up/down
- In some cases, no native mechanism for NSM

NSM – Security Onion

- Created by Doug Burks in 2008
- Free and open source platform for intrusion detection, enterprise security monitoring, and log management
- Can be installed from an ISO image or packages on top of Ubuntu 16.04 in minutes
- Traditionally fed data from a TAP or SPAN port
- Integrated with the Elastic Stack for data enrichment, indexing, and visualization

https://securityonion.net/

NSM – Security Onion - Overview



https://securityonion.net/

NSM – Security Onion – Alert Data

HIDS/NIDS Alerts

- Wazuh/Snort/Suricata

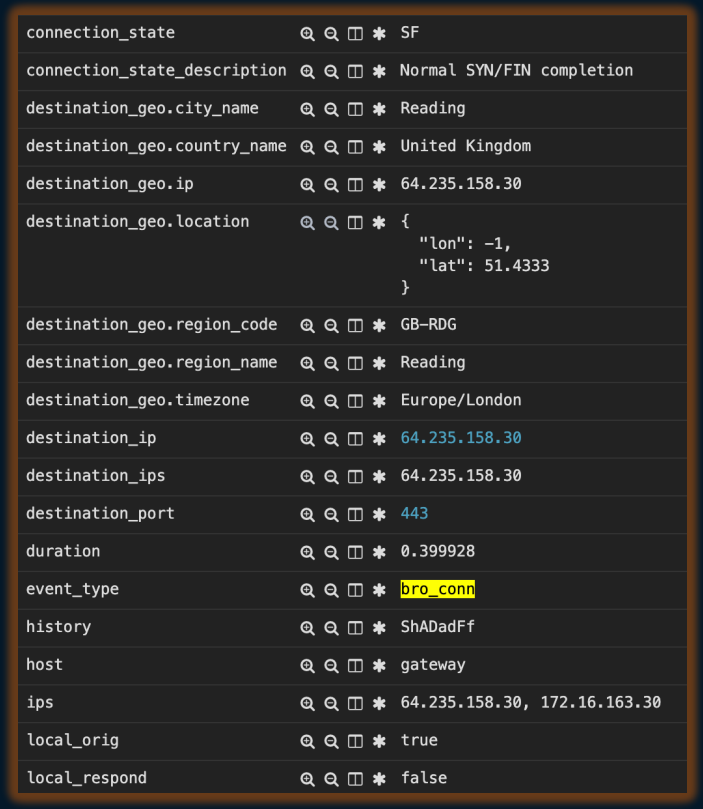
```
pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=127.0.0.1  user=wlambert
```

Alert	Source IP Address	Destination IP Address
ET TROJAN Generic - POST To .php w/Extended ASCII Characters (Likely Zeus Derivative)	192.168.3.35	188.124.5.100
ET TROJAN Zbot POST Request to C2	192.168.3.35	188.124.5.100
ET CURRENT_EVENTS Zbot Generic URI/Header Struct .bin	192.168.3.35	188.124.5.107
ET TROJAN JS/Nemucod requesting EXE payload 2016-02-01	192.168.3.35	188.124.9.56
SURICATA TCPv4 invalid checksum	188.124.9.56	192.168.3.35

NSM – Security Onion - Metadata

Network Metadata

- Bro/Zeek
  - DCE/RPC
  - DNS
  - HTTP
  - SSL
  - and much more

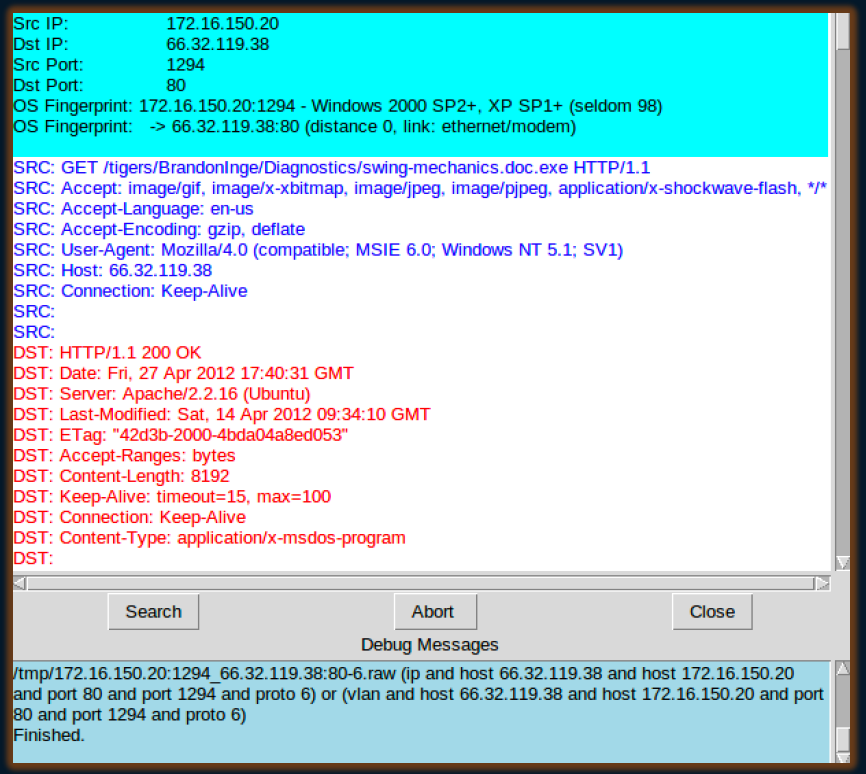


connection_state	SF
connection_state_description	Normal SYN/FIN completion
destination_geo.city_name	Reading
destination_geo.country_name	United Kingdom
destination_geo.ip	64.235.158.30
destination_geo.location	{ "lon": -1, "lat": 51.4333 }
destination_geo.region_code	GB-RDG
destination_geo.region_name	Reading
destination_geo.timezone	Europe/London
destination_ip	64.235.158.30
destination_ips	64.235.158.30
destination_port	443
duration	0.399928
event_type	bro_conn
history	ShADadFf
host	gateway
ips	64.235.158.30, 172.16.163.30
local_orig	true
local_respond	false

NSM – Security Onion – Full Content Data

Full Content Data (PCAP)

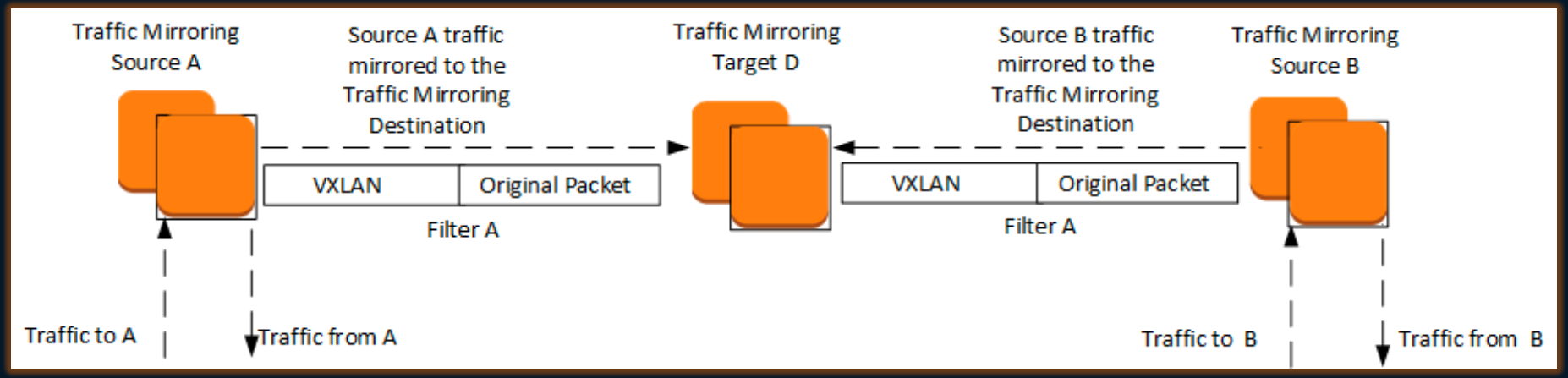
- Netsniff-NG
  - Complete transcript




aws

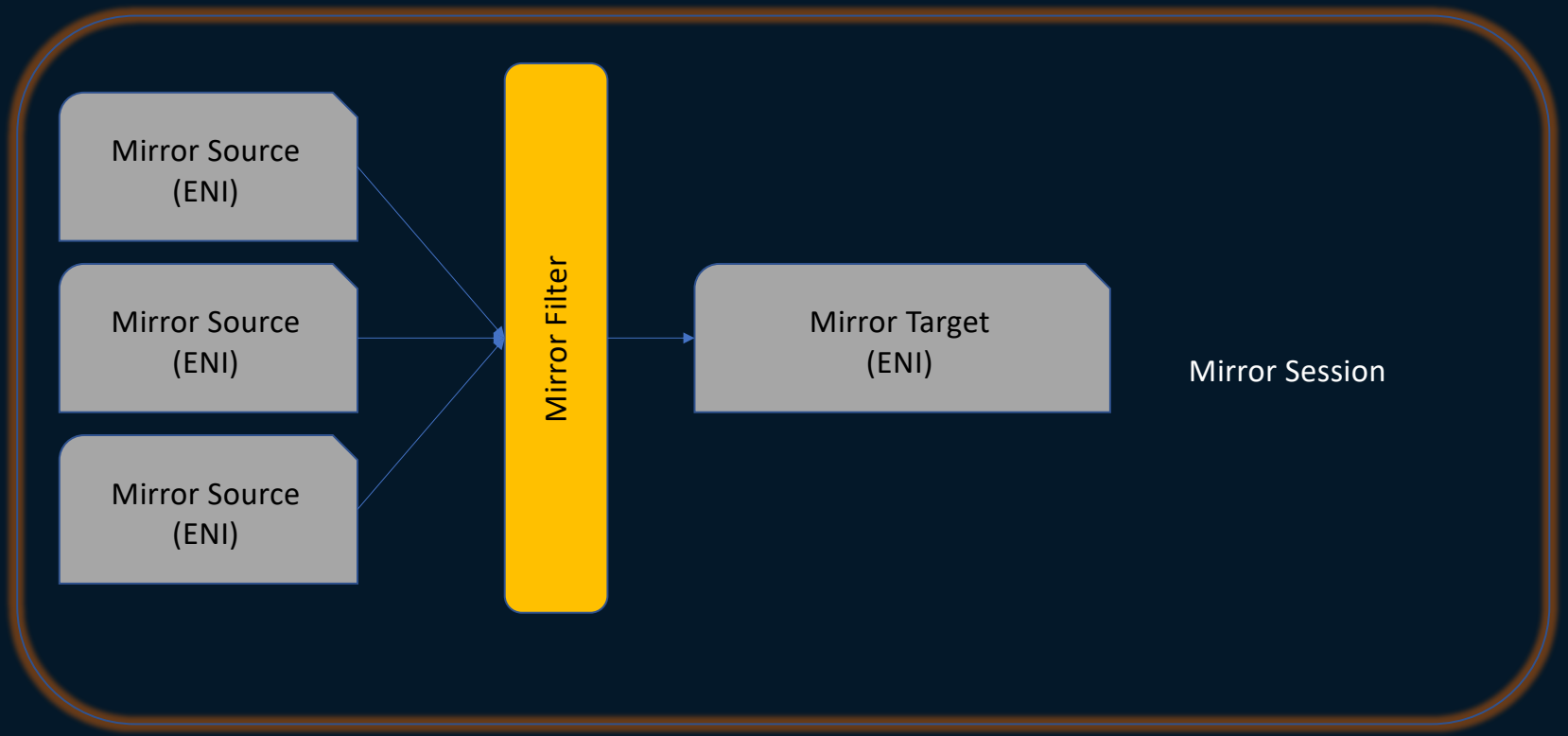
AWS – Native NSM Capability

- Traffic Mirroring
  - Mirror traffic from source network interface of EC2 instance to target interface
  - Packets encapsulated and transported via VXLAN



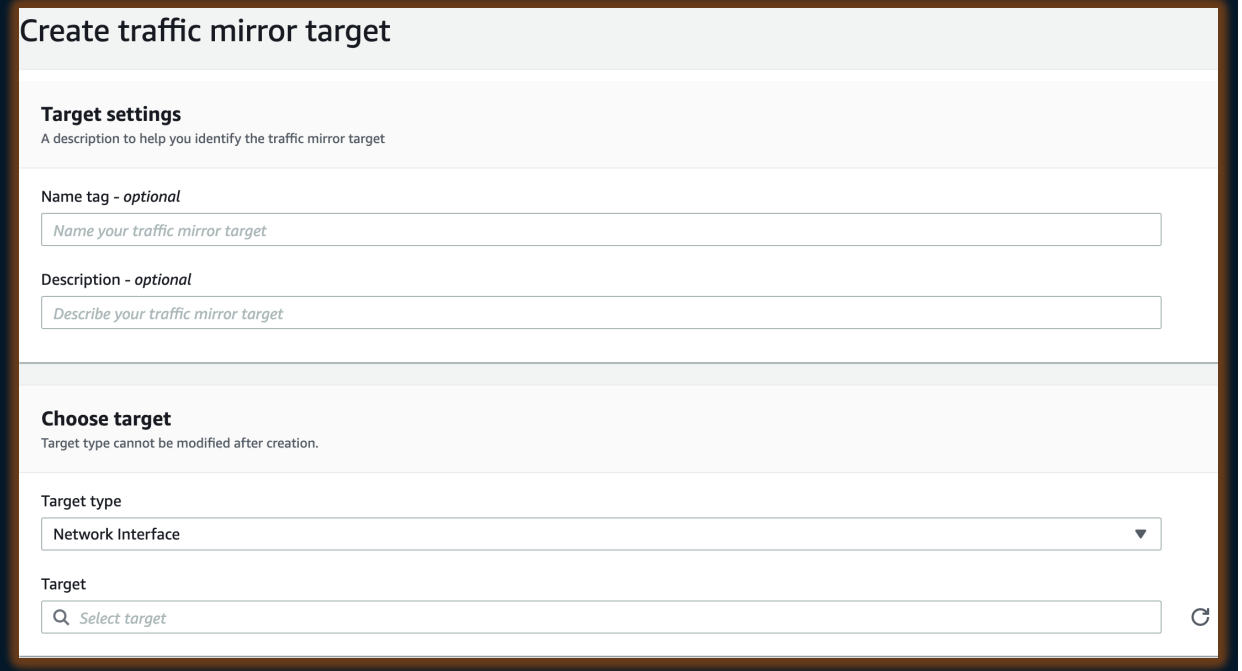
https://docs.aws.amazon.com/vpc/latest/mirroring/traffic-mirroring-how-it-works.html

AWS – Traffic Mirroring Components

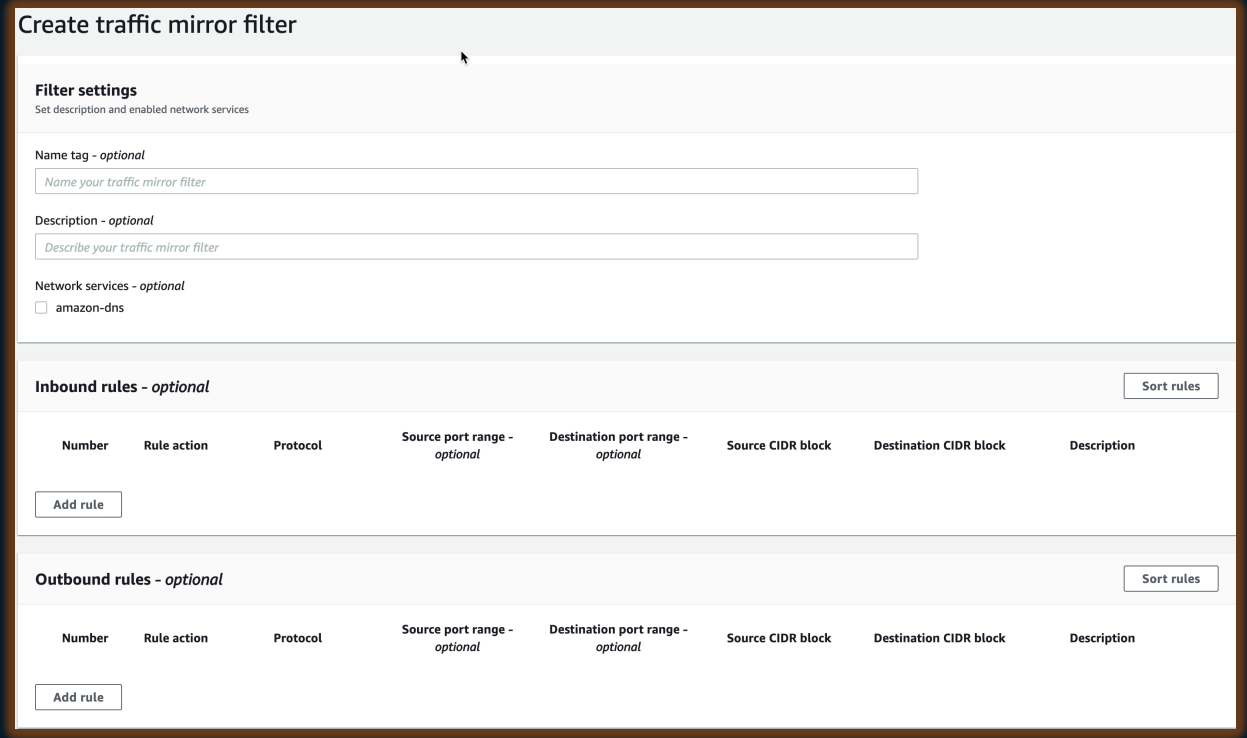


AWS – Traffic Mirror Target

Define the target interface where we would like to send mirrored traffic



AWS – Traffic Mirror Filter

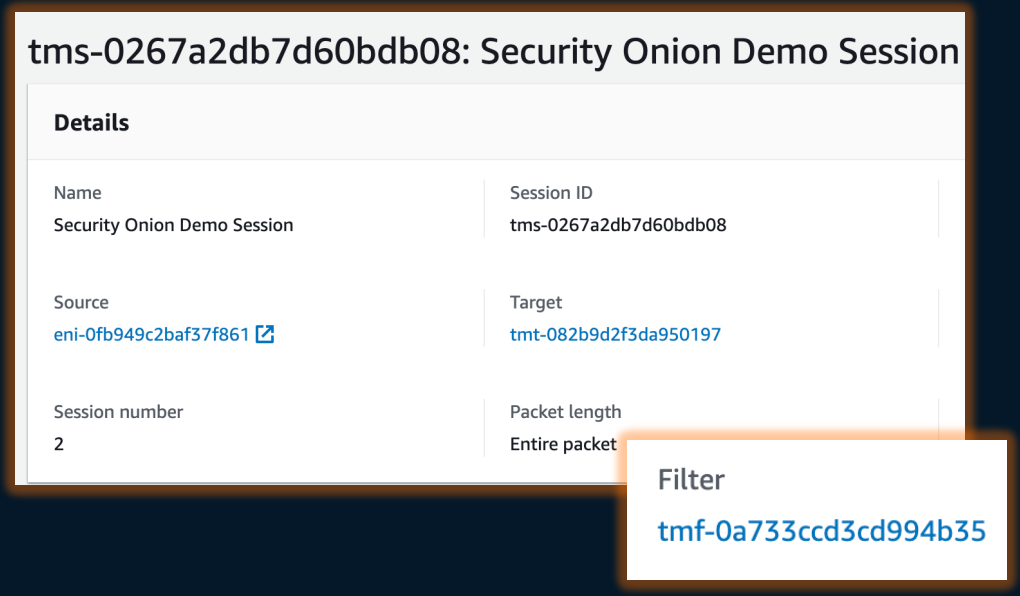


- Define the protocols/traffic to be mirrored
- Filter out any unwanted traffic

AWS – Traffic Mirror Session

Mirror Session is comprised of:

- Source interface (to be mirrored)
- Target Interface
- Mirror Filter



AWS - Cross-Account/Cross-VPC Mirroring

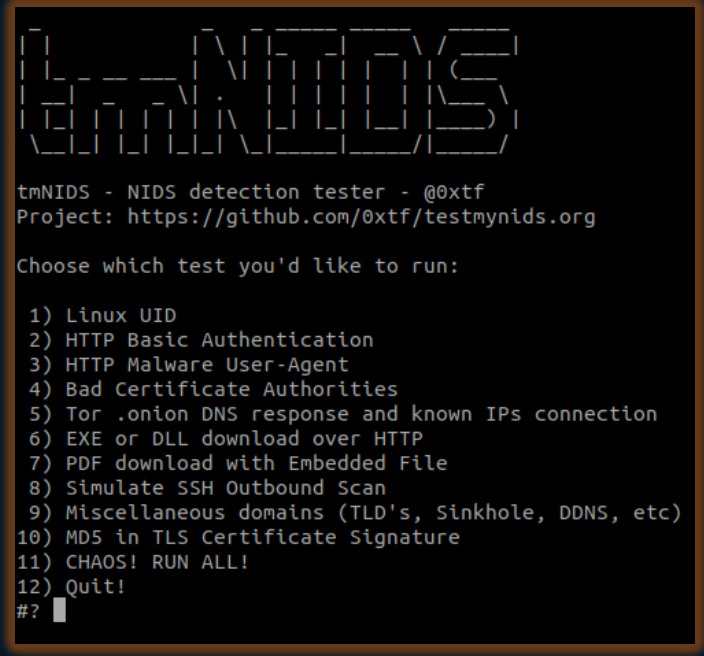
Source Owner	Source VPC	Source Target Type	Target Owner	Target VPC	Connectivity Option
Account A	VPC 1	Network interface	Account A	VPC1	No additional configuration
Account A	VPC 1	Network interface	Account A	VPC 2	Intra-Region peering or a transit gateway
Account A	VPC 1	Network interface	Account B	VPC 2	Cross-account Intra-Region peering or a transit gateway
Account A	VPC 1	Network interface	Account B	VPC 1	VPC sharing

To the Console!

tmNIDS

- Open-source
- Written by Tiago Faria, 3CoreSec (0xtf)
- Test alerting for cloud or on-premises NSM quickly and easily

ET SCAN Potential SSH Scan OUTBOUND
ET DNS Query for .su TLD (Soviet Union) Often Malware Related
ET DNS Reply Sinkhole - sinkhole.cert.pl 148.81.111.111
ET INFO DYNAMIC_DNS Query to a Suspicious no-ip Domain
ET POLICY DNS Query for TOR Hidden Domain .onion Accessible Via TOR
ET POLICY Outgoing Basic Auth Base64 HTTP Password detected unencrypted
ET TOR Known Tor Exit Node Traffic group 35
ET TOR Known Tor Exit Node Traffic group 75
ET TOR Known Tor Relay/Router (Not Exit) Node Traffic group 162
ET TOR Known Tor Relay/Router (Not Exit) Node Traffic group 227



3CS AutoMirror

- Open-source
- AWS Lambda function/serverless app
- Quickly and easily facilitate cloud NSM
- Monitors EC2 instances
  - If instances created with Mirror=True tag (and if instance is a supported type), their interfaces will be added to a VPC mirror session
  - Also works for instances that are tagged, then restarted

Google Cloud Platform

GCP – Native NSM Capability

- **Packet Mirroring**
  - Allows for capture of the complete conversation between hosts on the network
  - Extremely valuable for network forensics and security monitoring, and can be relatively inexpensive to implement compared to some other methods
  - Can specify subnets, network tags, or instance names

https://cloud.google.com/vpc/docs/packet-mirroring

GCP – Key Considerations

- You can mirror only TCP, UDP, and ICMP traffic.
- Mirrored sources and collector destination must be in the same region (can be in different VPCs w/ VPC Network Peering)
- Max number of mirror sources:
  - 5 subnets
  - 5 tags
  - 50 instances

https://cloud.google.com/vpc/docs/packet-mirroring#key_properties

GCP – Packet Mirroring Components

- Collector
- Mirrored source
- Internal load balancer and forwarding rule
- Packet mirroring policy
- VPC Peering if across different VPCs

GCP - Collector

- Instance(s) to which all traffic is mirrored
- Can be a group of instances, or a single instance
- Typically runs software that ”sniffs” the network traffic to look for anomalies
  - Ex. Snort/Suricata/Zeek
- In our case, the collector will be a Security Onion instance.

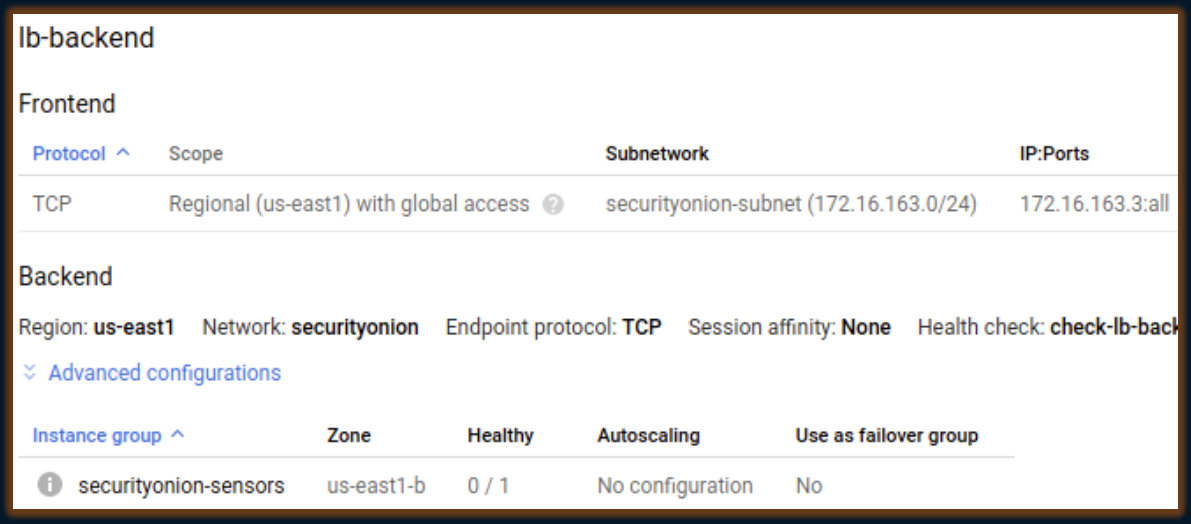
https://cloud.google.com/vpc/docs/using-packet-mirroring

GCP – Mirrored Source

- The interface(s)/instance(s) for which traffic will be mirrored
- Can be defined based off of:
  - Subnets
  - Tags
  - Name of instance(s)

https://cloud.google.com/vpc/docs/using-packet-mirroring

GCP – Internal Load Balancer



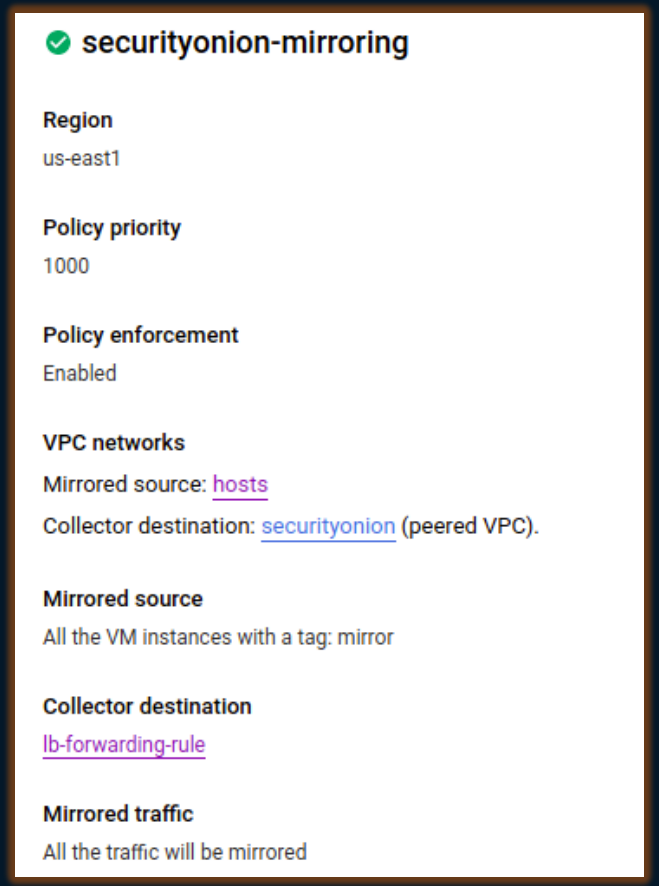
- Allows for routing of traffic from mirrored instances to collector instances behind the load balancer
- Must be in the same region as the instances to be mirrored
- Collector destination specified as an associated forwarding rule

https://cloud.google.com/vpc/docs/using-packet-mirroring#internal_load_balancer

GCP – Packet Mirroring Policy

- Region
- Priority
- VPC Networks
  - Mirrored Source
    - All VMs w/ mirror tag
  - Collector destination
    - lb-forwarding-rule
- Filtering
  - All traffic

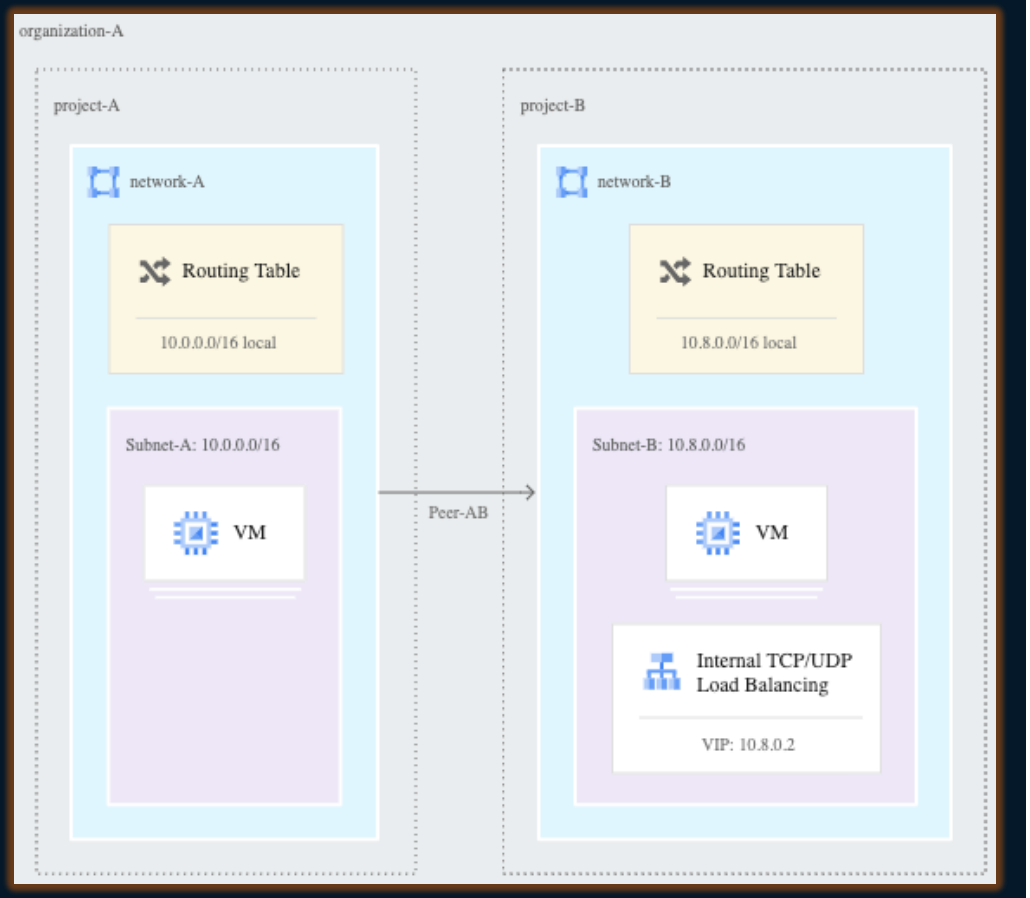
https://cloud.google.com/vpc/docs/using-packet-mirroring#creating



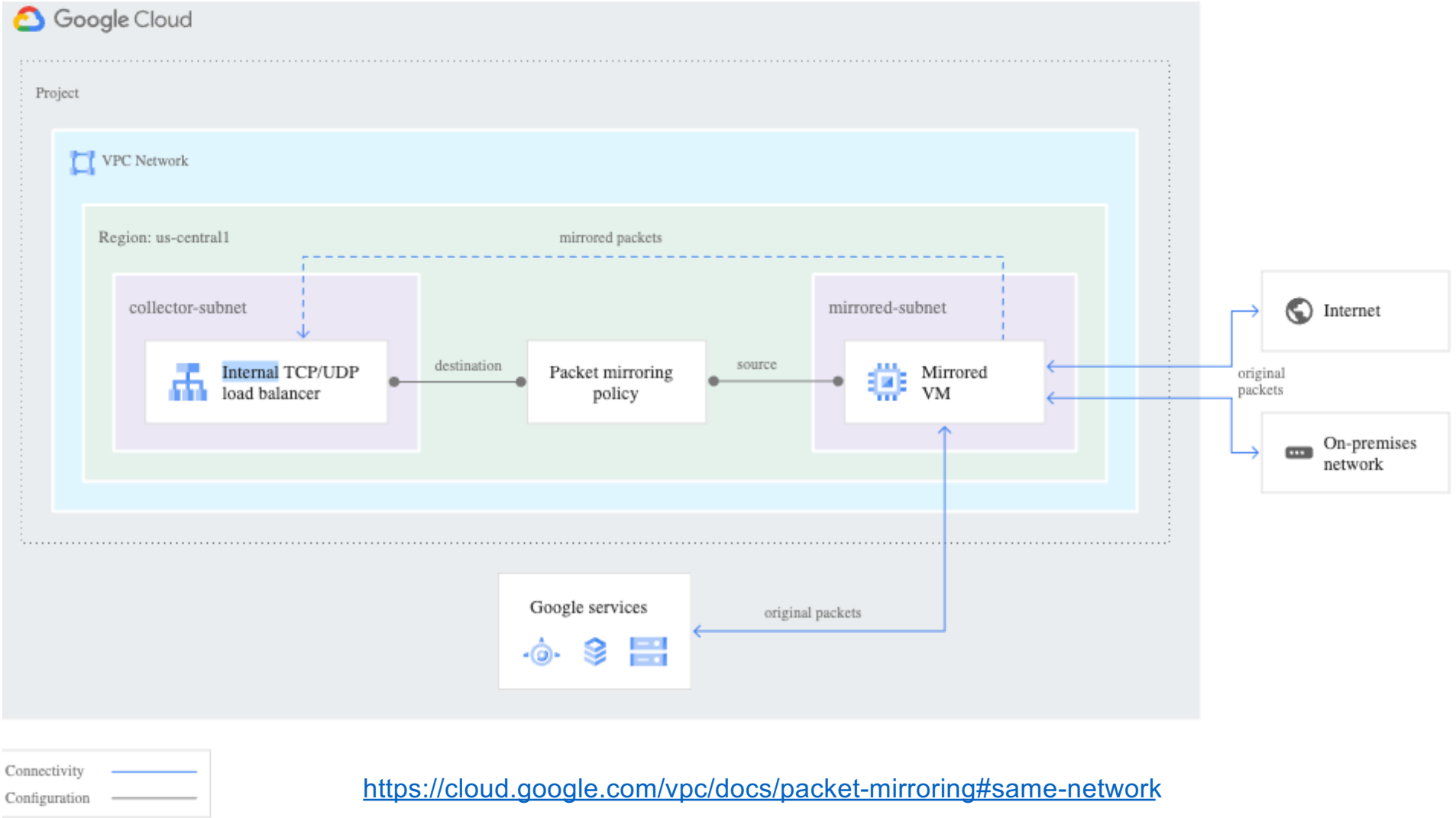
GCP – VPC Peering

Allow for cross—VPC mirroring

Name ↑	Your VPC network	Peered VPC network
peering1	hosts	securityonion
peering2	securityonion	hosts



https://cloud.google.com/vpc/docs/using-vpc-peering



https://cloud.google.com/vpc/docs/packet-mirroring#same-network

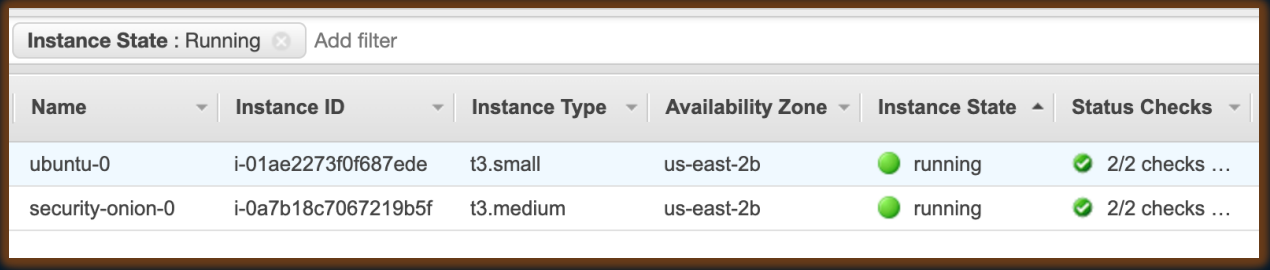
Automating NSM Deployment with Terraform

Terraform: NSM with AWS

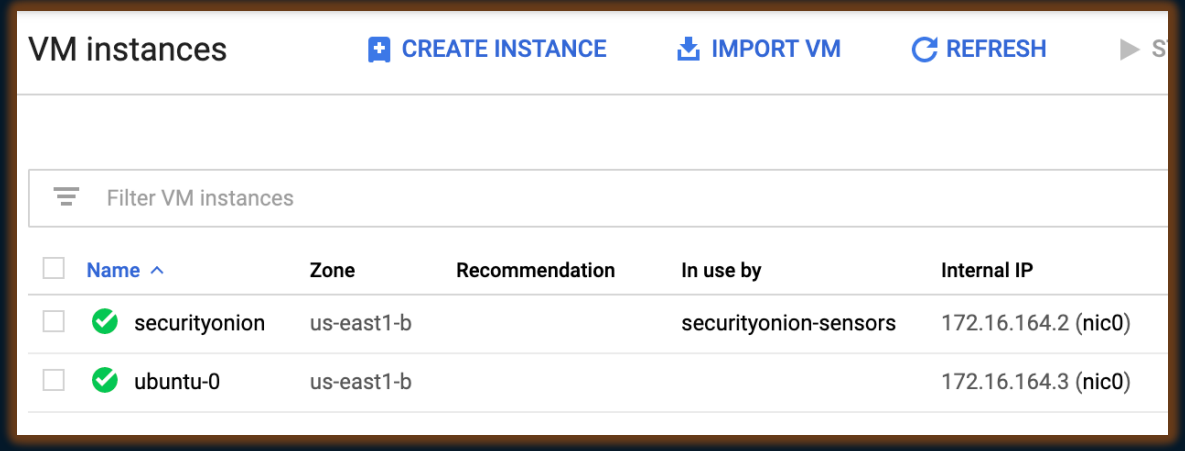
Follow the instructions below to setup a full-fledged VPC that includes:

- Security Onion
- Ubuntu/Windows hosts
- AutoMirror functionality to automatically mirror the traffic for each of the hosts

https://github.com/Security-Onion-Solutions/securityonion-cloud/tree/master/terraform



Terraform: NSM with GCP



https://github.com/Security-Onion-Solutions/securityonion-cloud/tree/master/terraform/gcp

vxlan2pcap

- Convert VXLAN-encapsulated PCAP to be readable by tools not supporting VXLAN encapsulation:

./vxlan2pcap vxlan.pcap out.pcap

```
wlambert@dev:~$ sudo tcpdump -nnr vxlan.pcap -c 3
reading from file vxlan.pcap, link-type EN10MB (Ethernet)
15:20:32.675392 IP 192.168.56.11.39924 > 192.168.56.12.4789: VXLAN, flags [I] (0x08), vni 123
ARP, Request who-has 10.0.0.2 tell 10.0.0.1, length 28
15:20:32.675732 IP 192.168.56.12.40908 > 192.168.56.11.4789: VXLAN, flags [I] (0x08), vni 123
ARP, Reply 10.0.0.2 is-at 4a:7f:01:3b:a2:71, length 28
15:20:32.676047 IP 192.168.56.11.48134 > 192.168.56.12.4789: VXLAN, flags [I] (0x08), vni 123
IP 10.0.0.1 > 10.0.0.2: ICMP echo request, id 3389, seq 1, length 64
wlambert@dev:~$ sudo tcpdump -nnr out.pcap -c 3
reading from file out.pcap, link-type EN10MB (Ethernet)
15:20:32.675392 ARP, Request who-has 10.0.0.2 tell 10.0.0.1, length 28
15:20:32.675732 ARP, Reply 10.0.0.2 is-at 4a:7f:01:3b:a2:71, length 28
15:20:32.676047 IP 10.0.0.1 > 10.0.0.2: ICMP echo request, id 3389, seq 1, length 64
```

https://github.com/Security-Onion-Solutions/securityonion-cloud/tree/dev/vxlan2pcap

FIN

@therealwlambert
@securityonion

Download Security Onion
http://securityonion.net

Download the RC1 Release of Security Onion 2.0
Ubuntu 8/Centos 7
Includes Sigma, Playbook, osquery, TheHive, Cortex, and more!
https://github.com/Security-Onion-Solutions/securityonion

Documentation:
https://securityonion.readthedocs.io

Professional Services, Training, and Support
https://securityonionsolutions.com

NSM Resources

Applied Network Security Monitoring – Chris Sanders, Jason Smith
Practice of Network Security Monitoring – Richard Betjlich