MSNETVIEWS: GEOGRAPHICALLY DISTRIBUTED MANAGEMENT OF ENTERPRISE NETWORK SECURITY POLICY

Iffat Anjum, Jessica Sokal, Hafiza Ramzah Rehman, Ben Weintraub, Ethan Leba, William Enck, Cristina Nita-Rotaru, Bradley Reaves

SACMAT 2023
Talk outline

- Motivation
- Zero Trust and Prior Work
- MSNetViews
- Evaluation
Talk outline

■ Motivation
■ Zero Trust and Prior Work
■ MSNetViews
■ Evaluation
Once upon a time...

- Networks were protected by secure perimeters
  - "Castle-and-moat defense"
Once upon a time...

- Networks were protected by secure perimeters
  - "Castle-and-moat defense"
- Users on the inside were trusted
Once upon a time...

- Networks were protected by secure perimeters
  - "Castle-and-moat defense"
- Users on the inside were trusted
- Users on the outside were not
Problem #1: Lateral movement

- Attackers had a hard time getting in
Problem #1: Lateral movement

- Attackers had a hard time getting in
- But once inside, became hard to contain
Problem #2: Distributed offices

- Enterprises no longer have their data or users in just one place
- Where should the perimeter be?
Problem #3: Advanced persistent threats

- An attacker may infiltrate a system on day one
Problem #3: Advanced persistent threats

- An attacker may infiltrate a system on day one
- But not move laterally until many days later
  - Makes detection difficult
These problems are real

- Colonial Pipeline temporarily halted all 5,500 miles of pipeline operations
- 45% of pipeline operators were affected
- 17 states declared a state of emergency
- Paid a ransom of $4.4M USD
Talk outline

- Motivation
- Zero Trust and Prior Work
- MSNetViews
- Evaluation
Zero trust

- Old paradigm
  - "Trust but verify"
Zero trust

- Old paradigm
  - "Trust but verify"
  - Analogous to checking ID when entering a bar
Zero trust

- **Old paradigm**
  - "Trust but verify"
  - Analogous to checking ID when entering a bar

- **Zero trust paradigm**
  - "Never trust, always verify"
  - Like checking ID when ordering each drink
Zero trust in practice

- Popularized by Google's BeyondCorp
- Critical services operate in cloud
- Multi-factor authentication
- Device attestation
- Behavioral analytics
ZT is nice in theory, **but ...**

- It's not possible to move everything to Cloud
  - Workstations
  - development/file servers,
  - device management interfaces
  - Etc.

- What about the on-premises network?
ZT is nice in theory, but ...

- It's not possible to move everything to Cloud
  - Workstations
  - development/file servers,
  - device management interfaces
  - Etc.

- What about the on-premises network?

In-network defenses are still needed
Prior work:
**NetViews**
(SACMAT '22)

- Addresses access control for non-cloud infrastructure
- Uses NGAC policy language
- Relies on SDN infrastructure
  - Flow rules enforce access control
- Does not address distributed enterprises
Addresses access control for non-cloud infrastructure

Uses NGAC policy language

Relies on SDN infrastructure

Flow rules enforce access control

Does not address distributed enterprises

Prior work: NetViews (SACMAT '22)
Enterprises with geographically distributed sites introduce new challenges...

- Users commonly move between sites
  - require differentiated access based on location
Enterprises with geographically distributed sites introduce new challenges...

- Users commonly move between sites
  - require differentiated **access** based on location
- Compromise of a single site should not leak the global policy

(a) Bank

(b) Big-Tech
Enterprises with geographically distributed sites introduce new challenges...

- Users commonly move between sites
  - require differentiated **access based on location**
- Compromise of a single site should **not leak the global policy**
- Only **site administrators should modify policies** for their local resources
Talk outline

- Motivation
- Zero Trust and Prior Work
- MSNetViews
- Evaluation
NGAC policies

- NGAC is a policy definition language
  - Defined by NIST in 2015
- Can model both ABAC and RBAC policies
- **Assignments** define hierarchy
- **Associations** define granted permissions
- **Prohibitions** define denied permissions

(Anjum et al., 2022)
Software-defined Networking (SDN)
Software-defined Networking (SDN)

- Network consists of
  - Devices
Software-defined Networking (SDN)

- Network consists of
  - Devices
  - Switches
Software-defined Networking (SDN)

- Network consists of
  - Devices
  - Switches
  - Controllers
Software-defined Networking (SDN)

- Network consists of
  - Devices
  - Switches
  - Controllers

- Controllers install flow rules on network switches
Software-defined Networking (SDN)

- Network consists of
  - Devices
  - Switches
  - Controllers
- Controllers install flow rules on network switches
- Switches use flow rules to route packets between devices and other switches
Overview of Multi-Site NetViews
Overview of Multi-Site NetViews

- Global policy management
Overview of Multi-Site NetViews

- Global policy management
- Site-local policy management
Overview of Multi-Site NetViews

- Global policy management
- Site-local policy management
- Policies defined with NGAC language
Overview of Multi-Site NetViews

- Global policy management
- Site-local policy management
- Policies defined with NGAC language
- Enforced by SDN flow rules
Overview of Multi-Site NetViews

- Global policy management
- Site-local policy management
- Policies defined with NGAC language
- Enforced by SDN flow rules
- Policy reacts to users roaming between sites
Overview of Multi-Site NetViews

- Global policy management
- Site-local policy management
- Policies defined with NGAC language
- Enforced by SDN flow rules
- Policies react to users roaming between sites
- Policy state is coordinated with a global manager
Policy Enforcement: Intent-based Networking

- Abstract "intent" from multiple flow rules
- Intents are compiled from NGAC policy
Roaming

- Users may move between sites
- User's access should be informed by location

Uses NGAC obligations – Dynamic, event-based policy elements

Creates assignments from users to location attributes

Detected locally at new site – Local manager informs global manager – Global manager informs the other sites
Roaming

- Users may move between sites
- User's access should be informed by location
- Uses NGAC obligations
  - Dynamic, event-based policy elements
Roaming

- Users may move between sites
- User's access should be informed by location
- Uses NGAC obligations
  - Dynamic, event-based policy elements
- Creates assignments from users to location attributes
Roaming

- Users may move between sites
- User's access should be informed by location
- Uses NGAC obligations
  - Dynamic, event-based policy elements
- Creates assignments from users to location attributes
- Detected locally at new site
  - Local manager informs global manager
  - Global manager informs the other sites
Policy slicing

- Global policy can leak confidential information about the organization
Policy slicing

- Global policy can leak confidential information about the organization
- Sites need not be aware of the local policies at other sites
Policy slicing

- Global policy can leak confidential information about the organization
- Sites need not be aware of the local policies at other sites
- Policies can be sliced on a "need-to-know" basis
- Slicing algorithm uses depth-first traversal to find relevant policy elements
Administrative Policies

- Defines what individual administrators can update in a policy
- Policy invariant rules to maintain policy semantics
- Leverages NGAC administrative policy semantics
Administrative Policies

- Defines what individual administrators can update in a policy
- Policy invariant rules to maintain policy semantics
- Leverages NGAC administrative policy semantics

For more details, please see the paper
Talk outline

- Motivation
- Zero Trust and Prior Work
- MSNetViews
- Evaluation
Experimental Setup

Compare

- Baseline (ONOS ifwd)
- NetViews
- MSNetViews

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total flows in MiniStanford Topology</td>
<td>1k</td>
</tr>
<tr>
<td>Total flows in Cisco Topology</td>
<td>32</td>
</tr>
<tr>
<td>Traffic pattern for experiments with 2 sites</td>
<td>site 1 → site 2</td>
</tr>
<tr>
<td>Wait between consecutive connections</td>
<td>100 ms</td>
</tr>
<tr>
<td>Same city latency (DC↔DC)</td>
<td>1 ms</td>
</tr>
<tr>
<td>Same region latency (DC↔NY)</td>
<td>11.2 ms</td>
</tr>
<tr>
<td>Global latency (DC↔CP)</td>
<td>105 ms</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Topology</th>
<th>Devices</th>
<th>Switches</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco [75]</td>
<td>12</td>
<td>10</td>
<td>Network of an enterprise with Cisco PIX firewall</td>
</tr>
<tr>
<td>MiniStanford [75]</td>
<td>100</td>
<td>25</td>
<td>Stanford backbone network</td>
</tr>
</tbody>
</table>
Throughput and Latency Results

(a) Cisco
(b) Ministantford

(scales differ for readability)
Throughput and Latency Results

(a) Cisco

(b) Ministanford

(scales differ for readability)
Throughput and Latency Results

MSNetViews overhead is negligible, particularly when sites are far apart.

(a) Cisco
(b) Ministanford

(scales differ for readability)
# Policy Update Performance

<table>
<thead>
<tr>
<th>Host No.</th>
<th>Policy Node No.</th>
<th>Average Delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Policy Checker</td>
</tr>
<tr>
<td>100</td>
<td>300</td>
<td>3</td>
</tr>
<tr>
<td>100</td>
<td>700</td>
<td>6</td>
</tr>
<tr>
<td>1000</td>
<td>3000</td>
<td>25</td>
</tr>
<tr>
<td>1000</td>
<td>7000</td>
<td>62</td>
</tr>
<tr>
<td>4000</td>
<td>12000</td>
<td>151</td>
</tr>
<tr>
<td>4000</td>
<td>28000</td>
<td>452</td>
</tr>
<tr>
<td>7000</td>
<td>21000</td>
<td>388</td>
</tr>
<tr>
<td>7000</td>
<td>49000</td>
<td>1153</td>
</tr>
<tr>
<td>10000</td>
<td>30000</td>
<td>654</td>
</tr>
<tr>
<td>10000</td>
<td>70000</td>
<td>2441</td>
</tr>
</tbody>
</table>

Table: Effect of Policy Graph Complexity on Average Policy Checking and Slicing Delay
Policy Update Performance

<table>
<thead>
<tr>
<th>Host No.</th>
<th>Policy Node No.</th>
<th>Average Delay (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Policy Checker</td>
</tr>
<tr>
<td>100</td>
<td>300</td>
<td>3</td>
</tr>
<tr>
<td>100</td>
<td>700</td>
<td>6</td>
</tr>
<tr>
<td>1000</td>
<td>3000</td>
<td>25</td>
</tr>
<tr>
<td>1000</td>
<td>7000</td>
<td>62</td>
</tr>
<tr>
<td>4000</td>
<td>12000</td>
<td>151</td>
</tr>
<tr>
<td>4000</td>
<td>28000</td>
<td>452</td>
</tr>
<tr>
<td>7000</td>
<td>21000</td>
<td>388</td>
</tr>
<tr>
<td>7000</td>
<td>49000</td>
<td>1153</td>
</tr>
<tr>
<td>10000</td>
<td>30000</td>
<td>654 (red)</td>
</tr>
<tr>
<td>10000</td>
<td>70000</td>
<td>2441 (red)</td>
</tr>
</tbody>
</table>

Table: Effect of Policy Graph Complexity on Average Policy Checking and Slicing Delay
Policy Update Performance

<table>
<thead>
<tr>
<th>Host No.</th>
<th>Policy Node No.</th>
<th>Average Delay (ms)</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Policy Checker</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>100</td>
<td>300</td>
<td></td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>100</td>
<td>700</td>
<td></td>
<td>25</td>
<td>38</td>
</tr>
<tr>
<td>1000</td>
<td>3000</td>
<td></td>
<td>62</td>
<td>81</td>
</tr>
<tr>
<td>1000</td>
<td>7000</td>
<td></td>
<td>151</td>
<td>189</td>
</tr>
<tr>
<td>4000</td>
<td>12000</td>
<td></td>
<td>452</td>
<td>516</td>
</tr>
<tr>
<td>4000</td>
<td>28000</td>
<td></td>
<td>388</td>
<td>428</td>
</tr>
<tr>
<td>7000</td>
<td>21000</td>
<td></td>
<td>1153</td>
<td>1024</td>
</tr>
<tr>
<td>7000</td>
<td>49000</td>
<td></td>
<td>654</td>
<td>688</td>
</tr>
<tr>
<td>10000</td>
<td>30000</td>
<td></td>
<td>2441</td>
<td>1883</td>
</tr>
<tr>
<td>10000</td>
<td>70000</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table: Effect of Policy Graph Complexity on Average Policy Checking and Slicing Delay

Figure: Effect of Number of Slices Needed to be Generated for Policy Updates.
Summary

- Zero trust is needed in today's enterprise network landscape
- MSNetViews solves problems of previous solutions
  - On-premises networks
  - Distributed sites
- MSNetViews addresses
  - Roaming
  - Policy slicing
  - Distributed administrative policies
- Performance comparable to single site setting

- Source code available: https://github.com/netviews/ms-netviews
- Paper available here:
MSNetViews: Backup Slides
Figure: Effect of number of roaming users and number of relevant sites on average location update time per user for users roaming globally (between WashingtonDC ↔ Copenhagen(CP)). Location update events are not batched.

(a) Location update time of one roaming user as a function of number of relevant sites

(b) Avg. location update time per user as a function of number of users roaming between two sites

(a) Batch Interval of 1 sec

(b) Batch Interval of 10 sec

Figure: Average location update time per user with batch processing at two different batch intervals as a function of number of users roaming globally (between WashingtonDC ↔ Copenhagen(CP))
### TABLE I: MSNetViews Policy Invariant Rules

<table>
<thead>
<tr>
<th>Rule</th>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Dangling PE</td>
<td>Each policy element must lead to at least one policy class.</td>
</tr>
<tr>
<td>2</td>
<td>Exclusive UA</td>
<td>Each user attribute must lead to only one policy class.</td>
</tr>
<tr>
<td>3</td>
<td>Exclusive OA</td>
<td>Each object attribute must lead to only one policy class.</td>
</tr>
<tr>
<td>4</td>
<td>Exclusive Associations</td>
<td>The source and target attributes of an association relation must lead to same policy class.</td>
</tr>
<tr>
<td>5</td>
<td>Exclusive Prohibitions</td>
<td>The source and target attributes of a prohibition relation must lead to same policy class.</td>
</tr>
<tr>
<td>No.</td>
<td>Requirement</td>
<td>MSNetViews Adherence</td>
</tr>
<tr>
<td>-----</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>1.</td>
<td>Enterprise assets have basic network connectivity</td>
<td>Yes</td>
</tr>
<tr>
<td>2.</td>
<td>The enterprise can observe all network traffic</td>
<td>Yes</td>
</tr>
<tr>
<td>3a.</td>
<td>The enterprise must be able to distinguish between what assets are owned or managed by the enterprise</td>
<td>Yes</td>
</tr>
<tr>
<td>3b.</td>
<td>The enterprise must be able to distinguish between the devices’ security postures</td>
<td>No</td>
</tr>
<tr>
<td>4.</td>
<td>Enterprise resources should not be reachable without accessing a PEP</td>
<td>Yes</td>
</tr>
<tr>
<td>5.</td>
<td>The data plane and control plane are logically separate</td>
<td>Yes</td>
</tr>
<tr>
<td>6.</td>
<td>Enterprise assets can reach the PEP component</td>
<td>Yes</td>
</tr>
<tr>
<td>7.</td>
<td>The PEP is the only component that accesses the policy administrator as part of a business flow</td>
<td>Yes</td>
</tr>
<tr>
<td>8.</td>
<td>Remote enterprise assets should be able to access enterprise resources without needing to traverse enterprise network infrastructure first</td>
<td>out-of-scope</td>
</tr>
<tr>
<td>9.</td>
<td>The infrastructure used to support the ZTA access decision process should be made scalable to account for changes in process load</td>
<td>Yes</td>
</tr>
<tr>
<td>10.</td>
<td>Enterprise assets may not be able to reach certain PEPs due to policy or observable factors</td>
<td>Yes</td>
</tr>
</tbody>
</table>