# Temporal Behavior in Network Traffic as a Basis for Insider Threat Detection

Brett Rajchel, John V. Monaco, Gurminder Singh
Naval Postgraduate School
Monterey, CA

Angela Hu
University of California, Los Angeles
Los Angeles, CA

Jarrod Shingleton, Thomas Anderson
US Army Network Enterprise Tech. Command
Monterey, CA

***Abstract*—Insider threats are a costly and dangerous problem for government and non-government organizations alike. Considering an insider's inherently privileged level of access on a network, the main principle of network defense—keep potential threats and outsiders out—does not apply to insider threats. Current defenses are largely based on the detection of insider threat indicators which are often manually compiled from past events. This approach is limited in scalability, has difficulty generalizing to new threats, and fails to consider the wide range of behaviors within an organization. In this work, we describe a system that detects potential insider threats through the characterization of temporal behavior on a network. Our approach is completely unsupervised, based on the assumption that there are many different behavioral norms within a network. After testing the system on an operational network with over 8,000 hosts, we show through a series of case studies that the approach is effective in detecting behavioral anomalies suitable for follow up by a human analyst.**

***Keywords—network traffic analysis, unsupervised machine learning, pattern-of-life, anomaly detection***

## I. Introduction

Insider threats are a costly and dangerous problem for government and non-government organizations alike. Modern organizations rely on oftentimes vast and spanning information networks to share mission critical information and conduct daily operations. Information networks are trusted to store sensitive, proprietary, and classified information, the unauthorized disclosure of which can lead to immensely expensive or even deadly consequences. With insider threats on the rise at the national level, information technology (IT) personnel find themselves at the forefront of a struggle to protect digital assets.

Considering an insider's inherently privileged level of access to an information network, the main principle of network defense—keep potential threats and outsiders out—does not apply to insider threats. Typical active defenses against insider threats include crude tripwires which rely on up-to-date insider threat indicators. While not without merit, these defenses can be easily avoided by a determined insider. Furthermore, recent and publicly available information on insider threats is scarce. Notwithstanding, insider threat activities are innumerable: they are as diverse as human behavior itself.

We describe and test a system that extracts network host and organizational behavior from network traffic and detects behavioral anomalies using unsupervised machine learning techniques for the purpose of insider threat detection. The system is developed and tested on an operational network with over 8,000 daily active hosts over a consecutive 6-day period.

The system takes into account the temporal behavior of each host on the network, capturing both circadian and packet inter-arrival characteristics. Under the assumption that there are a variety of behavioral norms within a network, we form clusters based on similarity among the hosts' temporal behaviors. Anomalous hosts are then detected based on two different metrics that capture the level of conformity with one of the clusters and movement between clusters over the course of observation. Through a series of case studies, we demonstrate that the system is able to effectively detect behavioral anomalies.

## II. Background

Anomaly detection for the purpose of cybersecurity has been thoroughly studied for over 30 years, dating back to Denning's statistical model for detecting network intrusions in 1987 [1]. Anomaly detection systems used for the purpose of detecting malicious network activity rely on having an accurate view of normal network activity in order to recognize and flag abnormal network activity [2]. Anomaly detection systems assume that malicious network activity has fundamentally abnormal characteristics not shared by normal or benign network traffic. Moreover, anomaly detection systems assume these inherent (and oftentimes unknown) abnormal characteristics or attributes can never be fully hidden or obfuscated despite an adversary's best attempts to do so.

Non-threatening insiders conform to an "organizational design" where they can be expected to behave in accordance with their organization's rules, social norms, and patterns. For this reason, experts agree that anomaly detection is an important factor in detecting insider threat activity [3]. Researchers have sought to tailor and operationalize anomaly detection systems to discover insider threat activity on computer networks. These detectors typically develop an understanding of normal host or organizational network activity and use this baseline to flag deviations from the baseline.

As pointed out by Sommers and Paxton in [2], anomaly detection systems typically generate a high number of false positives because they assume every deviation from normal is malicious and discount contextual factors that could describe these deviations as benign. Detection systems should seek to automate the process of finding insider threats to the furthest extent possible, thereby reducing the burden on human analysts, but not discount the need for them in the overall system.

Perhaps one of the earliest insider threat detection systems that leveraged anomaly detection was in 2009 when Cuputo et al. introduced Elicit. Elicit generated user-specific datasets by collecting information from a user's network traffic, digital object usage, and ancillary contextual information [4]. Their

978-1-7281-2547-3/20/$31.00 ©2020 IEEE

detector uses Bayesian networks to generate a threat score for a user's daily activity based on the probability of the activity being malicious—or indicative of an insider threat. Elicit collected and processed network traffic to create per-user "information-use events" corresponding to activities in which a user read, wrote, or printed documents stored on an organization's intranet [4]. Furthermore, Elicit gathered user events corresponding to browsing and searching. Elicit used additional contextual information including users' job titles, departments, and office locations to derive more meaning from information-user events, allowing Elicit to identify suspicious activity when a user deleted or copied a document maintained by a different department, for example [4]. Caputo et al.'s Bayesian network "encodes the probabilities of occurrence for each activity for benign and malicious users" based on organizational norms, which were determined with the help of insider threat subject matter experts [4].

The Defense Advanced Research Projects Agency (DARPA) project Anomaly Detection at Multiple Scales (ADAMS) further leveraged anomalies in large datasets in order to detect insider threats in U.S. Government networks [5]. ADAMS used machine learning techniques to identify anomalous activity. One of ADAMS's most significant contributions was identifying anomalies most typically generated by insider threats. The project confirmed through empirical data that, as expected, malicious insiders fetch more sensitive information than benign insiders, send more information outside of their organization, and are more active than benign insiders [5]. These observations paved the way for future generations of detectors by identifying and refining a set of confirmed insider threat indicators.

Legg et al. [6] acknowledge the need for robust automation and diverse data sources to detect anomalies associated with insider threats. In addition to other records, their system harvests email, web, computer access, and physical building access logs to construct "daily observation profiles." Their system assesses the profiles to generate alerts from both threshold-based anomalies and deviation-based anomalies. Legg et al. envisioned their system as part of a larger insider threat detection system where human analysts follow up with true and false positives, marking them as such to refine the parameters within the system and providing feedback to the system for continuous learning with a human in the loop.

Like previous work, we envision temporal behavior on a network to be incorporated with other modalities, such as computer system logs, personnel data, and physical movement within an organization. The technical solution proposed in this work does not attempt to remove humans from the system; instead it attempts to complement other means of countering insider threats. Where our work differs from prior work is in the flexibility of defining behavioral norms. Whereas most prior work has considered anomalies in reference to a single profile (either per-user or population wide), our approach considers how an individual aligns with and transitions between a large number of baseline behaviors. This is accomplished by first determining up to several dozen behavioral norms within the network and then considering how well a user aligns with and transitions between those norms.

## III. Feature Extraction and Anomaly Detection

Our proposed system is broken up into three components: feature extraction, in which temporal measurements are made on the timing of packets; clustering, in which hosts are grouped based on behavioral similarities; and anomaly detection, in which hosts are scored based on their conformity to each cluster and movement between clusters over the observation period.

### *A. Feature Extraction*

We describe two different feature sets: the first characterizes human behavior by measuring the volume of traffic as it occurs over each port and hour of the day; the second characterizes device behavior by measuring the distribution of packet inter-arrival times, again broken down by port. Our approach characterizes the behavior of each user over the course of 1 day and assumes a one-to-one mapping between users and hosts on the network, i.e. we use IP address as a proxy for user identity. We consider only hosts for which IP address does not change over a multi-day observation period, including wired devices with static addresses and wired devices with DHCP-managed addresses that remain assigned to the same device. This simplifying assumption enables a per-user characterization of temporal behavior but does not address multi-user machines on the network, an item we leave for future work.

#### *1) Human Behavior*

In regard to recognizing human activity patterns, we consider activity over the duration of a single day. In order to capture patterns at a finer level, we also consider activity within single hours. These time ranges allow us to characterize when a user is active holistically over the course of a day; from this information we can infer typical and atypical patterns given sufficient time to observe a user. More specifically, we can reasonably infer a user's typical circadian rhythm, e.g., when they log onto the network to start working, and when they take a break for lunch.

We count the number of packets each user sends in each hour of a given day. In order to characterize a user's activity type, we inspect the destination port of each packet sent by the user. We count packets sent to the same destination port during the same hour. Applying this measurement results in volumetric information broken down by destination port and hour of day. Instead of accounting for every possible destination port, we select and enumerate the most commonly used destination ports across the network. Hereafter, we refer to these as *time-of-day* (ToD) features, with an example shown in Fig. 1.

| | Port 443 | | | | … | Port 22 | | | |
|---|---|---|---|---|---|---|---|---|---|
| Host | Hour 0 | Hour 1 | … | Hour 23 | … | Hour 0 | Hour 1 | … | Hour 23 |
| Host A | 34 | 0 | … | 8088 | … | 0 | 0 | … | 997 |
| Host B | 0 | 4242 | … | 0 | … | 0 | 0 | … | 1212 |

Fig. 1. Example time-of-day features describing human behavior.

#### *2) Device Behavior*

To characterize device behavior, we aim to account for and differentiate between the potential high number of requests per time unit, driven by device activity, and the bursty network usage associated with humans [7]. The network services used as result of device activity will oftentimes in itself allow us to differentiate it from human activity. For example, connections to destination port 8014 for Symantec security updates are likely

the result of automated device activity. Like humans, devices have preferred destination ports as a result of how they were configured.

Instead of measuring volume over days and hours, as we did for the ToD features in Fig. 1, we measure volume within incremental time ranges for each considered port. That is, we count the number of packets transmitted by a device within a given time interval, measuring the time from when the previous packet was sent to when the next packet is sent. For example, if device A exhibits the activity demonstrated in Fig. 2, it would result in the binned feature vector also portrayed in Fig. 2. This histogram of packet interarrival times characterizes the rates at which upstream traffic is generated by a particular host.

| | | | | | | |
|---|---|---|---|---|---|---|
| Device A | Packet number | 1 | 2 | 3 | 4 | 5 |
| | Time sent (µs) | 1 | 3 | 4 | 7 | 8 |
| | Interevent time (µs) | | 2 | 1 | 3 | 1 |

| Device | (0-1] µs | (1-3] µs | (3-5] µs |
|---|---|---|---|
| Device A | 2 | 1 | 1 |

Fig. 2. Example transformation from packet time to time interval features.

In order to account for how devices typically transmit packets in a regular and high frequency fashion, the time range bins have to be small—we assess on the order of microseconds. Adding the measurements that were a result of our destination examination, our device feature vectors appear in their final form, as shown in Fig. 3. These are hereafter referred to as *time interval* (TI) features.

| | Port 8014 | | | | … | Port 22 | | | |
|---|---|---|---|---|---|---|---|---|---|
| Host | (0-1] µs | (1-3] µs | … | (45-60) mins | … | (0-1] µs | (1-3] µs | … | (45-60) mins |
| Host A | 1001 | 5005 | … | 0 | … | 19 | 98 | … | 0 |
| Host B | 58 | 607 | … | 0 | … | 6 | 4500 | … | 0 |

Fig. 3. Example time interval feature vectors describing device behavior.

### *B. Clustering*

We cluster hosts using K-Means, which utilizes randomly selected centroids to group samples and alternatively recomputes cluster centroids and cluster membership until convergence. Features are computed for each day of activity from each host, i.e., both the ToD and TI features characterize behavior over a single day. The number of clusters is determined separately for each feature type based on the maximum silhouette score. Silhouette score measures the extent to which points are actually clustered by comparing distances between samples sharing the same cluster membership with distances between samples in the next closest cluster [9].

### *C. Anomaly Detection*

While analyzing how the clusters themselves are formed describes organizational behavioral trends (to include anomalies), we are primarily interested in host behavior relative to individual baselines and macro-level network utilization trends for the purpose of insider threat detection. Here, we must take a closer look at cluster membership and how members are clustered differently over time. Of note, without a labeled dataset, declaring that a network host is demonstrating anomalous behavior is ultimately a subjective call until later confirmed by a human analyst. After network hosts are clustered across the observation period, we must quantitatively describe why a host is behaving normally or abnormally. Foremost, we note the cluster in which a host appeared on each day over a given timeframe and from this, determine the likelihood of a host following a particular cluster sequence.

The cluster each host belongs to on each day is determined by performing a K-Means clustering over the entire observation period. Fig. 4 shows an example of the cluster in which a host appeared on each day. From this, we calculate the number of unique clusters a host appears in and the number of times a host hopped between clusters. For example, Host C in Fig. 4 appears to be hopping between clusters; one might conclude they are behaving consistently due hopping between the same two clusters. Whereas in the case of Host D in Fig. 4, one might conclude this host is behaving abnormally due to membership in several different clusters. Denoting the number of hops and number of unique clusters, as shown in Fig. 4, describes these phenomena.

| Host | 8Feb Cluster | 9Feb Cluster | 10Feb Cluster | 11Feb Cluster | Days Active | Hops | Unique |
|---|---|---|---|---|---|---|---|
| Host C | 0 | 0 | 1 | 0 | 4 | 2 | 2 |
| Host D | 1 | 2 | NA | 4 | 3 | 2 | 3 |

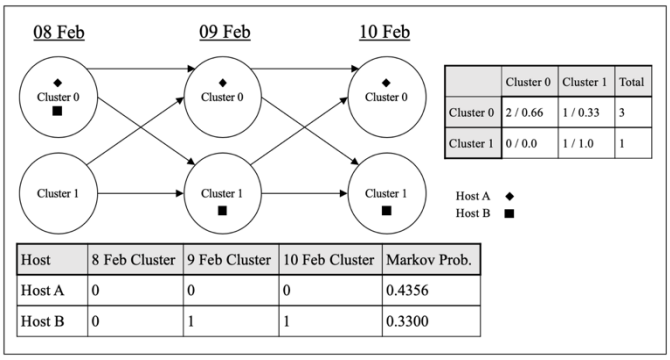
Fig. 4. Example clustering of each host on each day of observation.

We use the sample silhouette score as a metric to determine whether a host's behavior on a given day was appropriately clustered. In Fig. 4 for example, Host C's silhouette score on Feb. 8 (not shown) indicates it was likely appropriately placed in cluster 0, whereas Host D's silhouette score on Feb. 8 indicates it may have been inappropriately clustered in cluster 1 [9].

So far, we have focused on how network hosts are clustered relative to themselves. While this is useful for establishing per-host baselines, we need to address how hosts are clustered over time relative to other network hosts. For example, in isolation it may seem peculiar if a host appears in cluster 0 for four consecutive days, but then suddenly changes on the fifth day to cluster 6. However, this could be viewed as normal relative to the entire population if we observe a large proportion of the population also transition to cluster 6 on the same day. Such a massive distribution shift of hosts among each respective cluster is likely due to an outside factor, e.g., a mandatory online training event in which most users must participate or a shift in business working hours, and not an isolated change in host behavior. While understanding the specifics of such factors or events is not critical for anomaly detection, it is important to capture how individual hosts react to the imposition of these outside factors or events. Using our previous example, if a host does not shift to cluster 6, this could be cause for further investigation.

In order to reflect cluster distribution changes, and the significance of each host's movement between clusters across several days, we use a Markov chain to compute the probability of a host's particular cluster sequence. Over the observation period, a Markov chain captures the probabilities of all possible sequences of state changes; in our case each state is a cluster. We consider the union of all possible clusters in a given period and note the percentage of hosts whom transitioned from one state to another. This allows us to measure the likelihood—or the suspiciousness—of a host's cluster sequence. Fig. 5 shows an example in which two hosts are clustered over three

consecutive days. In this example, the likelihood of Host A's sequence (0 → 0 → 0) is greater than Host B (0 → 1 → 1).




| | Cluster 0 | Cluster 1 | Total |
|---|---|---|---|
| Cluster 0 | 2 / 0.66 | 1 / 0.33 | 3 |
| Cluster 1 | 0 / 0.0 | 1 / 1.0 | 1 |

| Host | 8 Feb Cluster | 9 Feb Cluster | 10 Feb Cluster | Markov Prob. |
|---|---|---|---|---|
| Host A | 0 | 0 | 0 | 0.4356 |
| Host B | 0 | 1 | 1 | 0.3300 |

Fig. 5. Example Markov chain and probabilities of cluster transitions.

This approach enables us to examine how a host is behaving relative to historical host trends and overall population trends. Examining the clusters themselves allows us to explain why a host or group of hosts is clustered in a given manner by describing the typical behavior demonstrated by the members of a given cluster. Given a sufficient observation period, our methodology allows us to derive both host-specific behavior baselines and the suspiciousness of a host's behavior relative to the population.

## IV. Results

### A. Data collection

We collected network traffic over 6 consecutive days from the core switch of a medium-sized university (3k students, 1k faculty/staff). Only the first 70 bytes of each packet were captured, as the features we defined are determined entirely by packet headers. The data collection resulted in approximately 1TB per day. Table 1 summarizes the number of active hosts over the contiguous 6-day collect: "DHCP Hosts" and "Static Hosts" show the number of active DHCP assigned or statically assigned IP addresses, respectively.

Table 1. Data collection overview

| Date | 8 Feb | 9 Feb | 10 Feb | 11 Feb | 12 Feb | 13 Feb |
|---|---|---|---|---|---|---|
| Day | Sat. | Sun. | Mon. | Tues. | Wed. | Thurs. |
| DHCP | 4,435 | 4,435 | 7,423 | 7,631 | 7,497 | 7,327 |
| Static | 479 | 498 | 490 | 484 | 519 | 526 |

The mix of week and weekend days allows us to infer network-wide and host-specific trends demonstrated on working and non-working days. Furthermore, 6 consecutive days of observation (8 through 13 Feb) allows the formation of at least partially representative host baselines. However, the lack of repeated same-day observations does not allow us to compare host-specific behavior demonstrated on specific days of the week. The system observed 8,838 unique IP addresses over the 6-day observation period; 537 were hosts with statically assigned addresses; 3,305 were hosts with DHCP wireless addresses; and 4,996 hosts had DHCP wired addresses. Fig. 6 shows a histogram counting the number of days each host was present on the network. Most wired DHCP and static hosts were active for all 6 days of the observation period, while most wireless hosts were active for only 4 days. Note that the observation period includes 4 business days and 2 non-business days (Saturday, Sunday), which suggests most wireless users do not visit campus over the weekend.

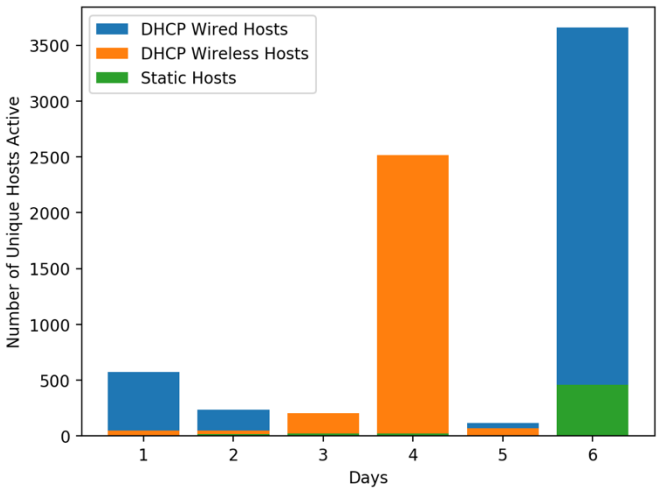



Fig. 6. Number of days a host was active, broken down by address type.

### B. Clustering Results

To start, we consider dimensionality reductions of time-based and port-based features to visually assess the separability of clusters within the dataset. There are 18 subnets within the network, some corresponding to building or department and some for designated devices, e.g., IP phones, printers, AWS, and administrative machines each have their own subnet. We color-code hosts after a t-distributed stochastic neighbor embedding (t-SNE), shown in Fig. 7. From this, we can observe that temporal behavior is largely different between devices from different subnets, and that some of the larger subnets are also separated into distinct clusters.

We use the silhouette score to determine the appropriate number of clusters separately for each feature type. Fig. 8 shows the average silhouette scores ranging from 4 to 150 clusters. The ToD features achieve a maximum score at 65 clusters, and the TI features at 27 clusters.

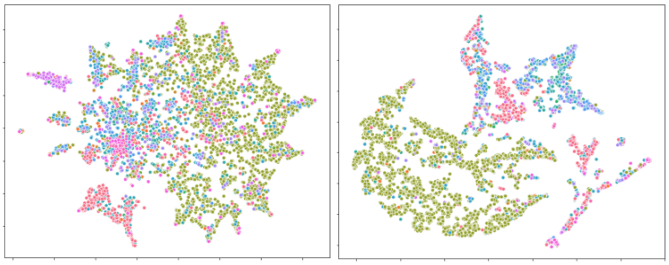


Fig. 7. t-SNE projections for hour (left) and port (right) features. Color denotes the subnet each host resides in.

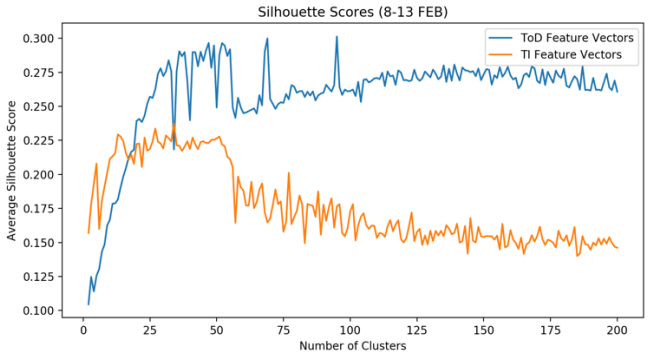



Fig. 8. Average silhouette score vs number of clusters for each feature type.

With the selected number of clusters for the time-of-day feature vectors being almost double the that was selected for the time interval feature vectors (65 for ToD vs 27 for TI), it is expected that the time-of-day hosts are more likely to occupy a higher number of clusters throughout the period. As shown in Fig. 9 (top left), the time-of-day hosts were most likely to be occupy 6 distinct clusters throughout the observation period, whereas time interval hosts were more likely to occupy 1 distinct cluster throughout the observation period.

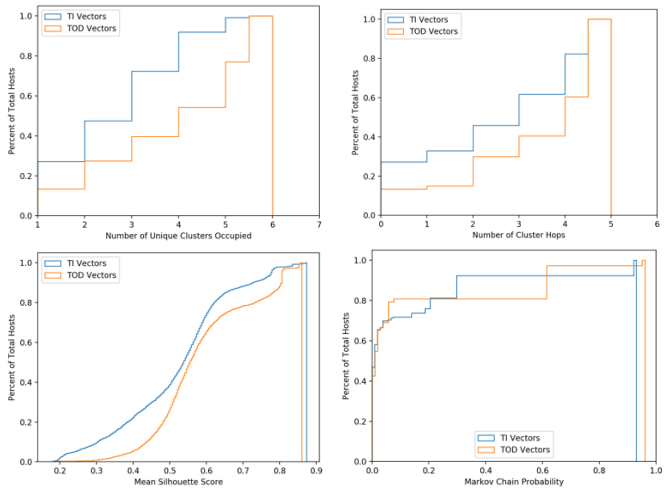



Fig. 9. CDFs of: clusters occupied (top left); cluster hops (top right); silhouette scores (bottom left) and Markov probabilities (bottom right). For ToD features the number of clusters k=65; for TI features, k=27.

Fig. 9 (top right) shows time-of-day hosts were most likely to change their cluster membership (cluster hop) the maximum amount of times (5 transitions for a period of 6 days), whereas time interval hosts were most likely to never hop clusters throughout the observation period. In isolation, the number of unique clusters a host occupies or the number of times a host changes cluster membership is nearly meaningless for the purpose of insider threat detection. A host's corresponding Markov chain probability and mean silhouette score computed over the observation period provide context which allows us to determine the significance of host cluster movement. For example, if not rationalized by population-relative metrics which characterize the population's behavior (in our case, Markov chain probabilities), a host changing cluster membership could be significant. Host-relative metrics could also help explain cluster movement. In this case, having host-unique baselines could help justify cluster movement if the movement is consistent with a host's past behavior. However, we assess our 6-day observation period is insufficient for developing host-specific behavior baselines. The lack of repeated same-day observations (i.e., multiple Monday observations) in our data and the data's brevity hinder our ability to build host-specific behavior baselines, an item we leave for future work. As a result, we will rely on the context provided by Markov chain probabilities and mean silhouette scores to drive the identification of behavioral anomalies.

Fig. 9 (bottom) shows the cumulative distributions of mean silhouette scores and Markov chain probabilities. Our system selects the hosts with the lowest scores (1%) as potentially anomalous and flags them for further investigation. The highest-scoring time-of-day hosts (in theory, demonstrating the least anomalous behavior) have a mean silhouette score of 0.3051 and a Markov chain probability of 9.64x10-7, respectively. The corresponding time interval hosts have a mean silhouette score of 0.1962 and a Markov chain probability of 5.17x10-8, respectively.

## C. *Case studies*

In the following case studies, we select and examine hosts the system has flagged as potentially anomalous. Specifically, we select one flagged host from each feature vector type for further investigation in an attempt to determine if they are indeed demonstrating anomalous behavior. This process follows a similar investigation as would be performed by a human analyst. The silhouette score and Markov chain probability are the first indicators that a host could be behaving anomalously relative to the population of network hosts. We augment these with a variety of techniques in the following case studies in order to provide further evidence that a flagged host is deviating from behavioral norms or not. Our techniques include host subnet identification and examination, the characterization of movement in and/or between clusters, and observing the raw network traffic generated by a host. We also consider per-hour activity and port usage for the time-of-day hosts and examine other characteristics as they appear relevant to an assessment of the host in question.

### *1) Time-of-Day Features – Host A*

Host A was flagged by the system due to its low Markov chain probability. As shown in Fig. 10, Host A earned the lowest Markov chain probability and a mean silhouette score of 0.4466, which is at the 14th percentile of all hosts flagged for low Markov chain probabilities. Of note, the next lowest Markov chain probability was 1.08x10-9 — significantly higher than Host A's.

| 8 – 13 Feb Clusters | Hops | Unique | Days Active | Mean SS | Markov Prob. | Type |
|---|---|---|---|---|---|---|
| 5 2 4 5 2 5 | 5 | 3 | 6 | 0.4466 | $4.08x10^{-11}$ | DHCP wired |
| Transition Probs. | 0.0014 | 0.0241 | 0.4516 | 0.0014 | 0.0018 | |

Fig. 10. Host A summary of activity, clusters occupied, silhouette score (SS) and Markov probability.

Examining Host A's cluster movement helps explain its low Markov chain probability. However, while other hosts had 5 hops and occupied 6 unique clusters, none had a lower Markov chain probability. From this we can conclude Host A made exceedingly unlikely cluster transitions across the observation period. Fig. 11 shows Host A's traffic volume patterns over each day, which are contradictory to the population. With the exception of Feb. 8 to 9 and Feb. 12 to 13, Host A's activity increases when the population mean decreases and vice versa. Particularly noticeable is Host A's divergence from the population on Feb. 9 and 10 where on Feb. 10 Host A's activity differs from the population mean by approximately 20,000 packets.

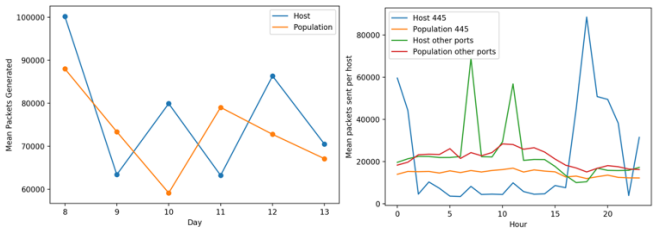



Fig. 11. Host A's traffic volume broken down by day (left) and hour/port (right) compared to the population.

While Fig. 11 (left) show's Host A's volumetric inconsistencies relative to the population, we must address the other metrics accounted for in the time-of-day features. Host A also appears to be behaving anomalously based on time metrics. Fig. 11 (right) shows Host A's per-hour activity across the entire observation period compared to the population mean. There are a few spikes in activity from Host A during hours 0, 1, 7, and 11. However, perhaps most striking is Host A's divergent activity between hours 17 and 21 — one of the population's lowest activity periods

Using the information gleaned from Fig. 11, we can examine Host A's activity on Feb. 10 during hours 17 through 21 compared to the population mean during the same hours. As shown in Fig. 12, (showing differences in per-hour activity between Feb. 9 and Feb. 10) the activity exhibited by Host A during hours 17 through 21 does significantly differ from the population mean. However, Fig. 12 also shows stark differences during other hours on Feb. 9 and 10. Furthermore, Fig. 12 reveals potential intra-host anomalies in Host A's behavior. Without a host-specific behavioral baseline to refer to with repeated same-day observations, we cannot claim this is a host-relative behavioral anomaly; despite this, Host A's significant decrease in traffic from hours 8 through 17 followed by a sudden spike in traffic volume could be cause for further investigation.

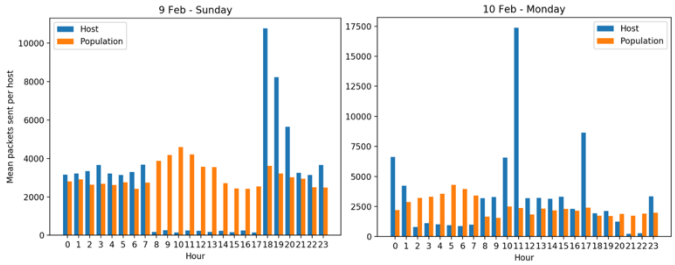



Fig. 12. Host A's hourly traffic volume compared to the population.

Host A divergences concerning port usage are less stark. This reveals a possible deficiency in our methodology given the previously mentioned anomalies. Specifically, we enumerated the 12 most commonly used ports, binning uncommon ports into a bulky "other" column. Expanding the "other" ports into separate hour/port columns could have resulted in more telling results. However, a closer examination of Host A's most frequently utilized ports—binned "other" ports and 445—compared to the population indicates that Host A uses port 445 significantly more.

We have shown that Host A, at times, demonstrated anomalous behavior relative to the population. To account for the diversity of device types (servers, printers, phones, etc.) in the population, we also consider Host A relative to other hosts in its subnet. We observed activity from at least 460 other hosts in Host A's subnet; 12 of the 40 hosts flagged by the system for low Markov chain probabilities were members of Host A's subnet. Based on this information, it is possible an unusual subnet-wide event prompted the subnet to behave unusually during the observation period. Fig. 13 helps explain Host A's anomalous activity, showing synchronized activity changes with the exception of Feb. 12 to 13. Fig. 13 also shows correlated changes in activity on an hourly basis between Host A and the subnet mean. The similar activity shifts between consecutive hours and consecutive days could indicate a shared pattern or event that influences activities over the entire subnet, including Host A.

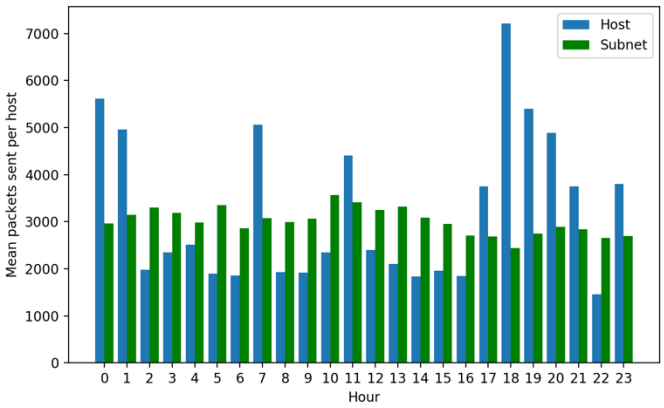



Fig. 13. Host A's hourly traffic voluem compared to it's subnet.

The preceding results do not explain the greater volume demonstrated by Host A compared to the subnet mean—particularly between hours 17 and 21. Also, Host A has the resoundingly lowest Markov chain probability and has a mean silhouette score in the 26th percentile of all subnet hosts. Host A's port usage could help explain this, as well as draw a distinction between the subnet's behavior. As shown in Fig. 14, while port 445 and port 443 usage are similar, Host A has significantly more "other" port activity, and less port 22 activity.

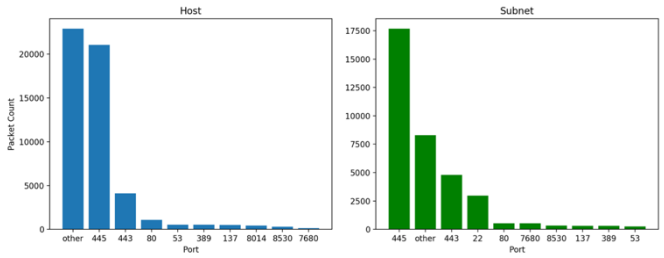



Fig. 14. Host A's port activity compared to the subnet.

We have shown that Host A's behavior is significantly different from the population of network hosts during the observation period. Activity levels (based on volume) between Host A and the population contradict one another on a daily and hourly basis. We have also shown that while Host A and its subnet have mostly corresponding daily and hourly activity patterns, Host A's port utilization and volume is significantly different from its subnet.

*2) Time Interval Features – Host B*

Host B was flagged by the system due to its low mean silhouette score. Host B has the lowest mean silhouette score of all the flagged time interval hosts. As seen in Fig. 15, Host B appeared in the same cluster on all days in the observation period, resulting in zero cluster hops. Despite Host B's lack of cluster movement, its Markov chain probability appears below the mean (0.4230) of other hosts that had zero cluster hops. Of note, Host B consistently appears in a large cluster that does not appear to be very well defined. For additional context, we were provided information that Host B is in a subnet used for energy management controllers associated with heating, ventilation, and air conditioning on campus. We observed activity from 57 other hosts in this subnet. These 57 hosts appear in the same cluster more than any other cluster.

| 8 – 13 Feb Clusters | Hops | Unique | Days Active | Mean SS | Markov Prob. | Type |
|---|---|---|---|---|---|---|
| 6 6 6 6 6 6 | 0 | 1 | 6 | 0.1742 | 0.3019 | DHCP wired |
| Transition Probs. | 0.7864 | 0.7864 | 0.7864 | 0.7864 | 0.7864 | |

Fig. 15. Host B summary of activity, clusters occupied, silhouette score (SS) and Markov probability.

Given this context and the unique nature of Host B, we might expect its behavior to be vastly different from the behavior of the population. Furthermore, given the cluster's below average definition, we should expect more variability in behavior from hosts in that cluster. For these reasons, it is unlikely that comparisons between Host B and the population, or Host B and other hosts in the cluster will be useful from an insider threat perspective. However, this likely indicates that Host B is indeed demonstrating anomalous behavior with respect to the population and the cluster.

As suspected, Host B's behavior significantly differs from the population. Fig. 16 shows Host B is significantly less active than the population average. It also shows that the top-5 packet time intervals do not intersect with those of the population. Of note, Host B's top-5 packet time intervals are all larger than the population's top-5 intervals. Regarding port utilization, Host B exclusively uses ports outside of the top ports we enumerated. We find that the average host in the same cluster is more similar to the population's average host than Host B—demonstrating a similar activity level and sharing the same top-2 packet time intervals as the population.

While not accounted for in the time-interval feature vectors, we are able to find similarities in the hourly activity patterns between Host B and the population. As already shown, the activity levels relative to volume are significantly different, but Host B and the population are both least active during hour 18 as shown in Fig. 17. These results suggest that time-of-day behavior alone may not be sufficient to discover anomalous temporal behavior, with characterization of time interval densities also playing an important role.

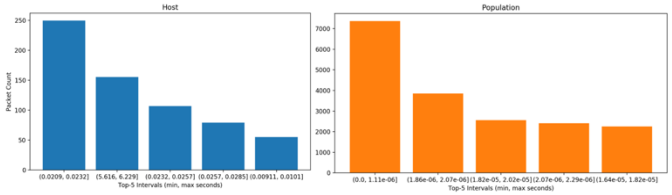



Fig. 16. Host B top-5 time intervals compared to the population.

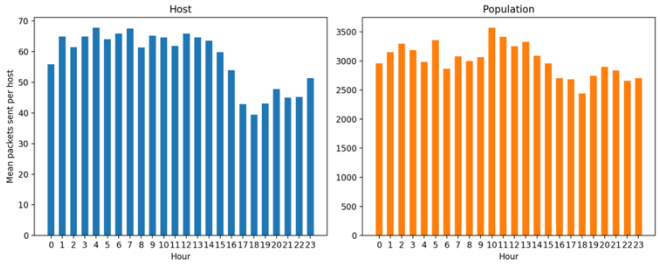



Fig. 17. Host B hourly activity compared to the population.

Despite some similarities in Host B's per-hour activity, it is clear that Host B demonstrated anomalous behavior relative to the population. And based on its mean silhouette score, it is also an outlier in respect to its cluster. However, in this case, Host B's identity as an energy management controller partly explains why it is behaving differently and why it was subsequently flagged by our system. Given it its unique nature, we would have to rely on direct comparisons with other subnet members or itself (host-relative) to determine if it is perhaps an insider threat.

Fig. 18 helps us understand the differences in behavior from Feb. 8 to 9. Considering the otherwise low activity levels on the 9th, the subnet's mean per-hour activity spikes during hours 10 and 14. While this could be cause to investigate other hosts in the subnet, it does not show Host B is behaving anomalously relative to its subnet.

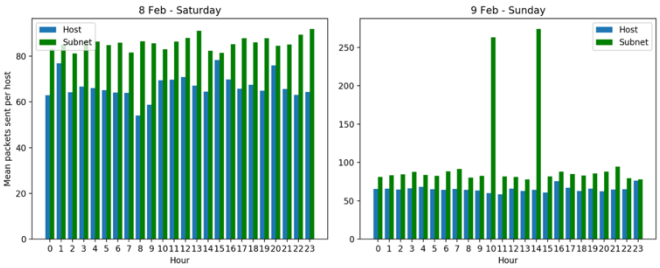



Fig. 18. Host B hourly traffic volume on Feb. 8 and 9.

Like Host A, we have shown that Host B's behavior is significantly different from the population of network hosts during the observation period. Comparing Host B's behavior with the behavior of other hosts in its cluster partly explains its low mean silhouette score and why it was flagged by the system. Host B's unique role in the network as an energy management controller explains its anomalous behavior relative to the population and its cluster. However, we identified several behavior differences between Host B and its subnet. First, Host B consistently demonstrates less per-day and per-hour activity. And second, Host B communicates with different interevent periods. In the processes of examining Host B we found a spike in activity from members of Host B's subnet on Feb. 9 (Sunday)—a day where we would generally expect less activity relative to other days.

Host B is indeed behaving anomalously relative to the population and perhaps relative to its subnet. Without additional operational context and information, we cannot conclude Host B is a strong candidate to investigate further as a possible insider threat. Given Host B uniqueness, comparisons with the population are likely less meaningful. A host-relative approach for identifying behavioral anomalies would likely be more effective for Host B.

## V. Conclusion

As demonstrated in the case studies, each feature type captures different aspects of host behavior, with ToD features aiming to characterize user activities with coarse granularity and TI features describing device behavior at a finer scale. As a result, we were able to recognize different behavior anomalies based on feature vector type. Anomalies based on time-of-day features are more comprehensive and intuitive for a human analyst, compared to time interval features, which are designed for the detection of device (automated) behavioral anomalies at a granular level. Given the mix of human and automated activity on many hosts (making it difficult to separate the two), we believe this feature vector type would be beneficial for host fingerprinting and the subsequent development of host-relative baselines, which this research largely did not address.

The case studies did not fully enumerate the advantages and disadvantages of using Markov chain probabilities versus silhouette scores for behavioral anomaly detection. However, the case studies did point out one important point for consideration: hosts flagged for having the lowest silhouette scores had very little cluster movement—most occupied 1 unique cluster and had 0 hops. Additionally, they usually were consistently clustered day-to-day on the periphery of a given cluster. This explains why (as was the case for Host B), the hosts flagged for their low silhouette scores were unique relative to the population. This is favorable for behavioral anomaly detection. However, we suspect these hosts demonstrated this consistent abnormal behavior because of their unique roles on the network and not because they were insider threats. Host-relative metrics could be used to evaluate these hosts for insider threat activity.

While mean silhouette scores appeared to be more effective in detecting repeated abnormal population-relative behavior, the Markov chain probabilities appeared to be effective in detecting behavioral anomalies as a result of dynamic or new emerging behavior. As seen with Host A, the Markov chain was useful in detecting changes in behavior that were different from the population (i.e., "spikes"). If we presume behavioral anomalies as a result of behavior change are indicative of insider threats, the Markov chain probabilities are likely more useful for our purposes.

In production, a network is likely to experience changes over time that could significantly alter what is considered normal behavior. This may include, for example, organizational changes or the introduction of a new tool or service. Because clusters have been determined *a priori* over the entire observation period, a production system would need to adapt to dynamic network conditions to avoid concept drift, for example by periodically updating cluster centers or limiting cluster analysis to shorter time periods.

The analysis of network traffic will continue to play an important role in insider threat detection, increasingly so for virtual organizations [10]. Future work will focus on the establishment of host-specific baselines over longer periods of observation. The separation of network traffic induced by human and automated device processes also remains a significant challenge, which would enable modeling each component separately.

## References


[1] D. E. Denning, "An Intrusion-Detection Model," IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987, doi: 10.1109/TSE.1987.232894.

[2] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network Intrusion detection," in 2010 IEEE Symposium on Security and Privacy, Oakland, CA, USA, 2010, pp. 305–316, doi: 10.1109/SP.2010.25.

[3] W. T. Young, A. Memory, H. G. Goldberg, and T. E. Senator, "Detecting unknown insider threat scenarios," in 2014 IEEE Security and Privacy Workshops, May 2014, pp. 277–288, doi: 10.1109/SPW.2014.42.

[4] D. Caputo, M. Maloof, and G. Stephens, "Detecting insider theft of trade secrets," IEEE Secur. Priv., vol. 7, no. 6, pp. 14–21, Nov. 2009, doi: 10.1109/MSP.2009.110.

[5] Defense Advanced Research Projects Agency, "Final report for the DARPA ADAMS project," College Park, MD, USA, 2015. [Online]. Available: https://apps.dtic.mil/dtic/tr/fulltext/u2/a625184.pdf.

[6] P. A. Legg, O. Buckley, M. Goldsmith, and S. Creese, "Automated insider threat detection system using user and role-based profile assessment," IEEE Syst. J., vol. 11, no. 2, pp. 503–512, Jun. 2017, doi: 10.1109/JSYST.2015.2438442.

[7] B. Gonçalves and J. J. Ramasco, "Human dynamics revealed through web analytics," Phys. Rev. E, vol. 78, no. 2, p. 026123, Aug. 2008, doi: 10.1103/PhysRevE.78.026123.

[8] F. Radicchi, "Human activity in the web," Phys. Rev. E, vol. 80, no. 2, p. 026118, Aug. 2009, doi: 10.1103/PhysRevE.80.026118.

[9] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," J. Comput. Appl. Math., vol. 20, pp. 53–65, Nov. 1987, doi: 10.1016/0377-0427(87)90125-7.

[10] Liu, Liu, et al. "Detecting and preventing cyber insider threats: A survey." *IEEE Communications Surveys & Tutorials* 20.2 (2018): 1397-1417.