Rochester Institute of Technology

Heated Alert Triage (HeAT) – Network Agnostic Extraction of Cyber Attack Campaigns

Stephen Moskal and Shanchieh Jay Yang

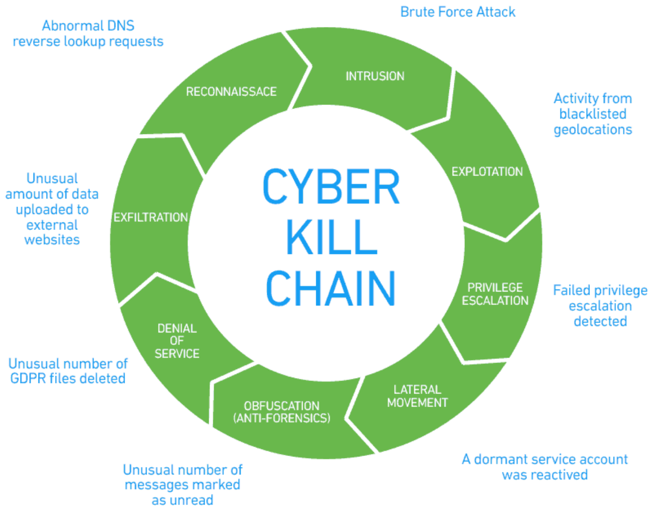
Ph.D. Candidate

MS in Computer Engineering

Rochester Institute of Technology

- **Motivation:** Cyber attack behaviors extremely diverse and complex
  - The actions performed by the adversary is dependent on the network infrastructure and the skill set of the adversary.
    - Hypothesis: the same "attack" on different networks may have similar characteristics but conducted differently due to the network infrastructure.

- **Ideally:** Use a labelled dataset describing the attack stages for various attackers, scenarios, and networks. Train a model
  - Does not exist -- attacks are constantly evolving
  - IDS's are inaccurate and produce overwhelming amounts of data – time is limited for analysts



How do we extract out the "kill chain" given the IDS alert logs?

Instead: *Use a limited amount of labelled data and leverage unsupervised/semi-supervised ML techniques along with feature engineering to extract attack scenario charateristics*

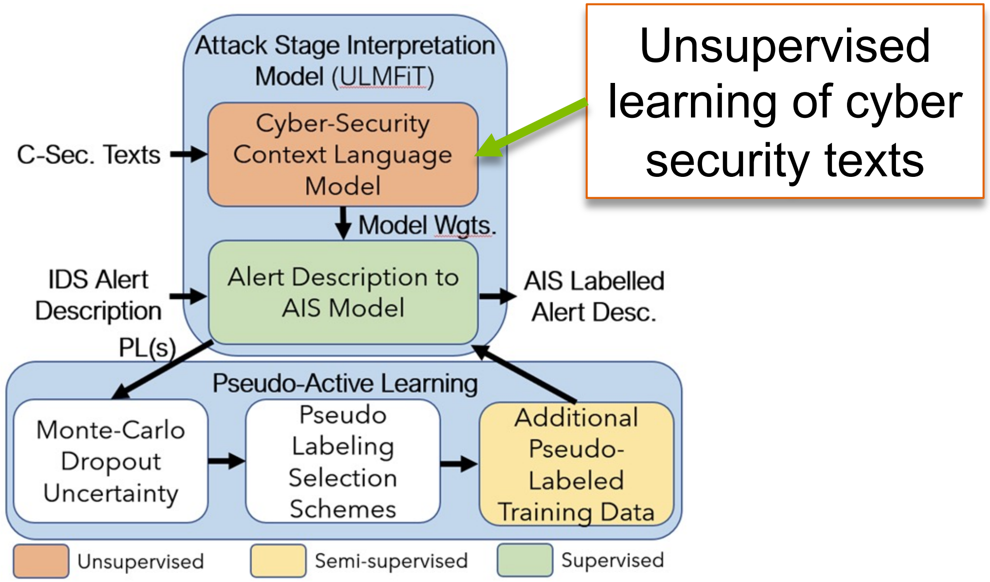
We ask: **If adversarial activity is known to have occurred on a network**, can we:

1) *leverage the IDS alert logs to extract out the relevant alerts pertaining to the adversarial actions*, and

2) *describe the attack campaign as a set of concise and intuitive "stages" so that campaigns can be compared*

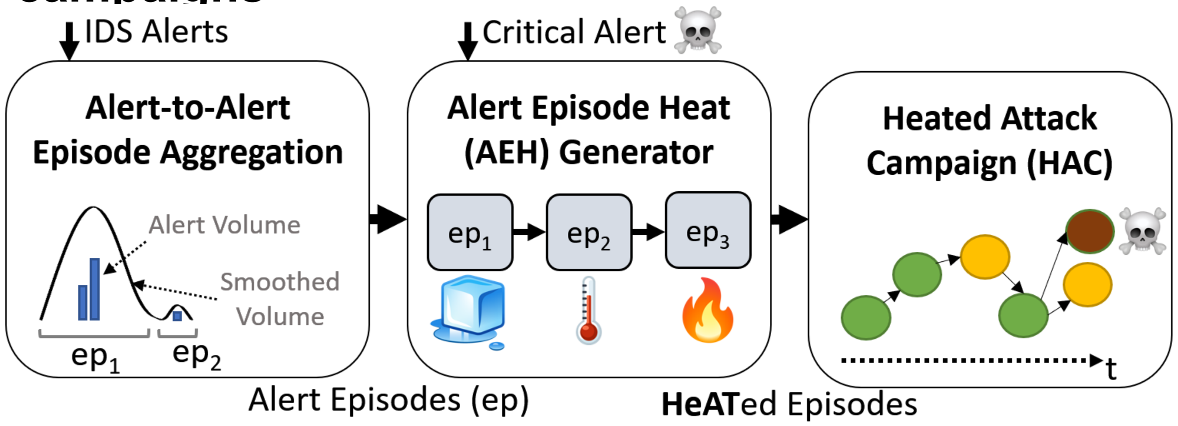
Remember: SOC analyst's time and resources are extremely limited **<- Our solution should not be a burden either!**

PATRL – Semi-supervised process to determine the **attack stage** (kill-chain like) of *any* IDS alert signature (Deep-NLP)



\+

HeAT – Use **prior network triage's** to create a network-agnostic model to **uncover other attack campaigns**

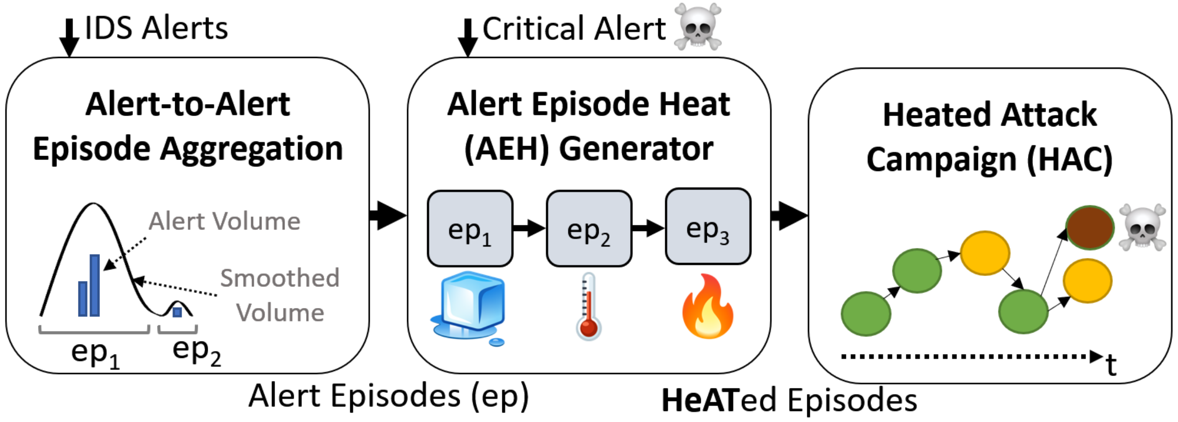


Heated Alert Triage (HeAT) – Network Agnostic Extraction of Cyber Attack Campaigns

- Problem: **Trace** the steps/**stages** an attacker took to compromise a network (attk. campaign (AC)) given some **critical IoC** (Indicator of Compromise)
  - SOC analysts triage IDS logs for other evidence (e.g. recon scans, asset exploitation) to determine if an IoC is a legitimate threat
  - Analysts have their own **knowledge of the network, prior observations, and cyber-expertise**!

Can we capture this assessment to explain other campaigns?

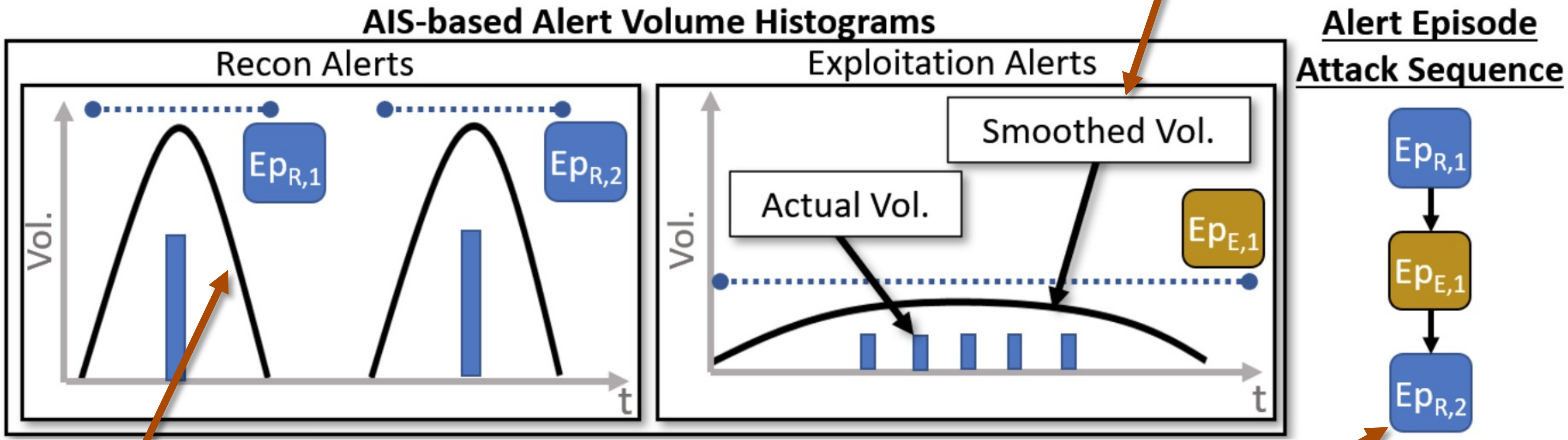
- Approach:
  - Alert Episode Heat – Ranks (0-3) how an `episode` of alerts contributes to the AC of a critical IoC – Attack-stage based
  - Network-Agnostic Features- Determine AC characteristics with no specific network info – **Apply HeAT to other adversaries and networks**!
  - HeATed Attack Campaign- Concise representation of the attack stages conducted by the attacker in time – **Respond to threats quickly**



Dealing With High Volume of IDS Alerts – "Alert Episodes"

- **Problem**: IDS's produce an overwhelming amount of alerts per-day (~10k-1M)
  - Often many false positives or one 'action' causing many alerts (recon, scripting, etc.)
  - **Objective:** Consolidate similar alerts based on the attack type, IP addresses, and time

Method: Apply Gaussian Smoothing to the volumes of alerts for each source IP, for each attack stage



The 'decline' in alert volume signifies the end of an action

Episodes represent similar alerts that are likely to be caused by one action by an adversary

Core HeAT Concepts

Alert Episode Heat (AEH)

- Given an IoC, AEH represents the contribution of a prior event to the IoC's attack campaign

AEH	Description
0	No relation to critical event
1	Recon. actions that may provide info. about e_c
2	Exploitation of assets giving access required to achieve e_c
3	Exfiltration/DoS/Access to info. directly relevant to e_c

Higher heat = Significant progress towards IoC

1. Perform a short triage using IoC's found on the network
2. Apply AEH values to prior events w.r.t the IoC
3. Use network-agnostic features train a model so that other scenarios can be realized given other IoC's

Network Agnostic Features

- Engineered features of the relation between two episodes with no specific network info

Name	Symbol	Description
Ep. Peak	e_{peak}	Time of peak alert volume
Ep. Start	e_{start}	Time of earliest alert
Ep. End	e_{end}	Time of latest alert
Distinct Source(s)	e_{src}	Sc
Distinct Target(s)	e_{tgt}	Sc
Distinct Sig(s)	e_{sig}	Sc
Distinct Dest. Port(s)	e_{port}	Sc
AIS	e_{ais}	A

Type	Feature	Description
Time	Ep. Interval Overlap	Overlap between the start & end times of e_c and e_p
	Ep. Peak Time Diff.	$e_{c,peak} - e_{p,peak}$
	Ep. Start Time Diff.	$e_{c,start} - e_{p,start}$
	Ep End Time Diff.	$e_{c,end} - e_{p,end}$
IP	Has Matching Source	1 if $e_{c,src} \cap e_{p,src}$ else 0
	Has Matching Target	1 if $e_{c,tgt} \cap e_{p,tgt}$ else 0
	Matching Source Ratio	Ratio of matching source IPs
	Matching Target Ratio	Ratio of matching target IPs
	Crit. Source as Target	1 if $e_{c,src} \cap e_{p,tgt}$ else 0
	Crit. Target as Source	1 if $e_{c,tgt} \cap e_{p,src}$ else 0
Action	Critical Ep. AIS	1-hot encoded $e_{c,AIS}$
	Prior Ep. AIS	1-hot encoded $e_{p,AIS}$
	Has Matching Sigs.	1 if $e_{c,sig} \cap e_{p,sig}$ else 0
	Matched Sig. Ratio	Ratio of matching signatures
	Matching Dest. Port	1 if $e_{c,port} \cap e_{p,port}$ else 0

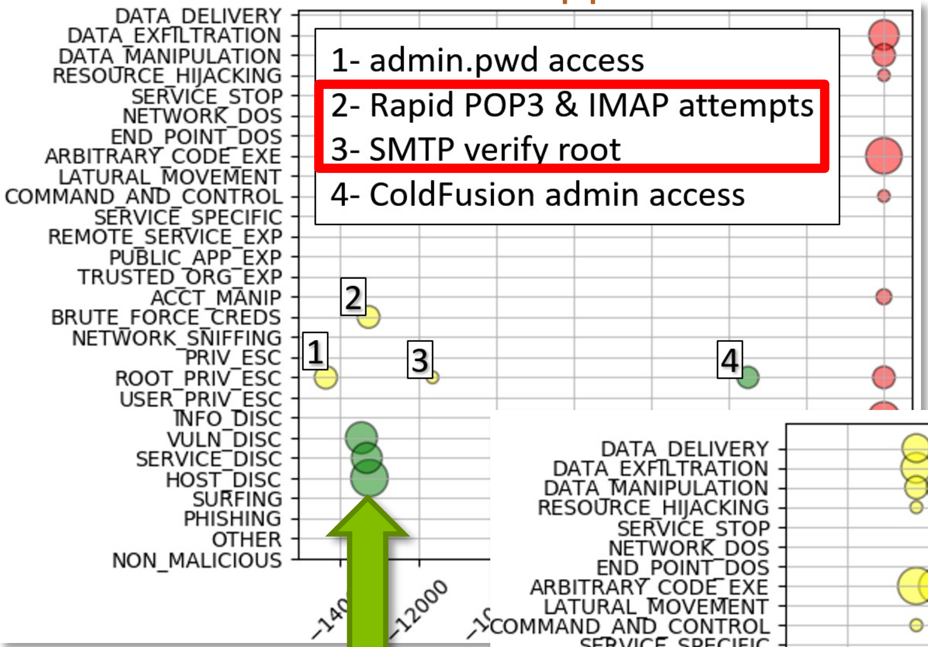
Attributes such as IP, timestamp, etc. are **specific to a single network**

These features enable us to **characterize** the indicators of an attack and use them to **uncover other scenarios**

HeATed Attack Campaign Examples – “CodeRed”

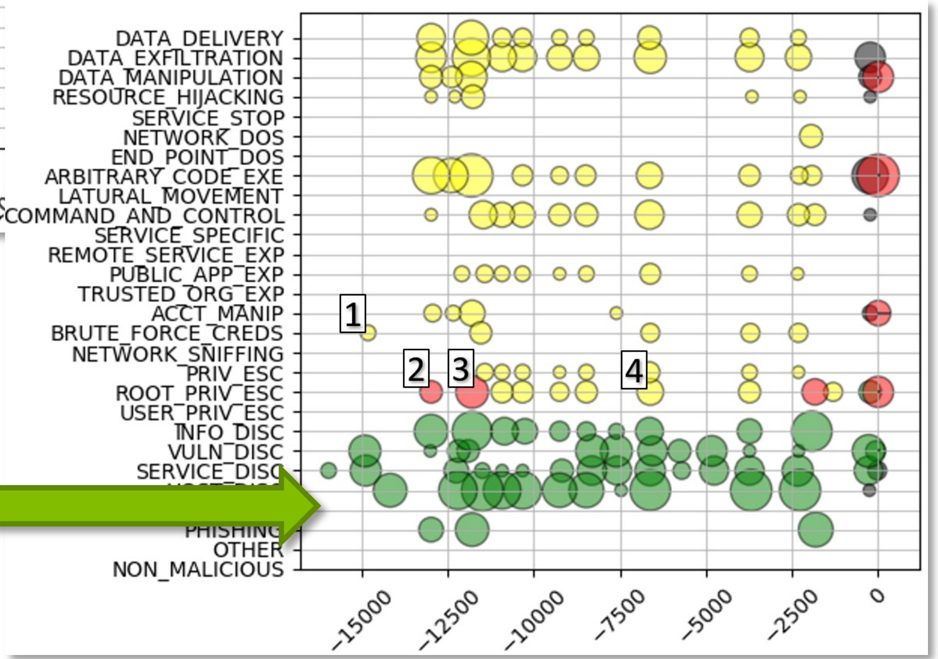
HeATing Different Adversaries (CPTC18)

“Calculated” Approach



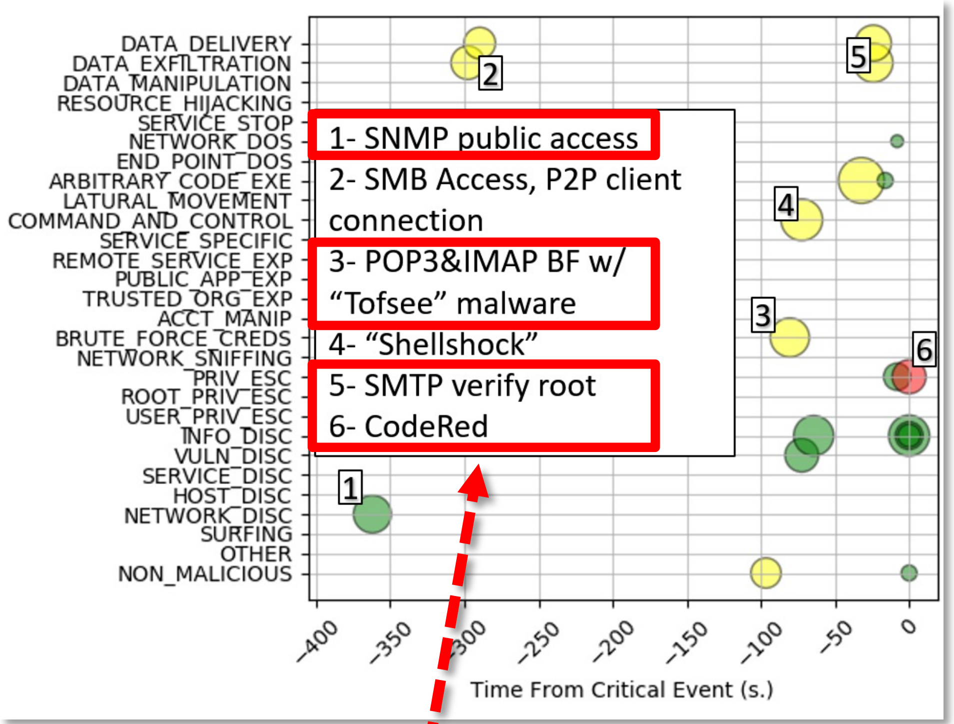
Same critical IoC & network, very different behaviors!

“The Script Kiddie”



10,000’s of alerts represented as a handful of alert episodes

HeATing Different Networks (CCDC18 w/ CPTC observations)



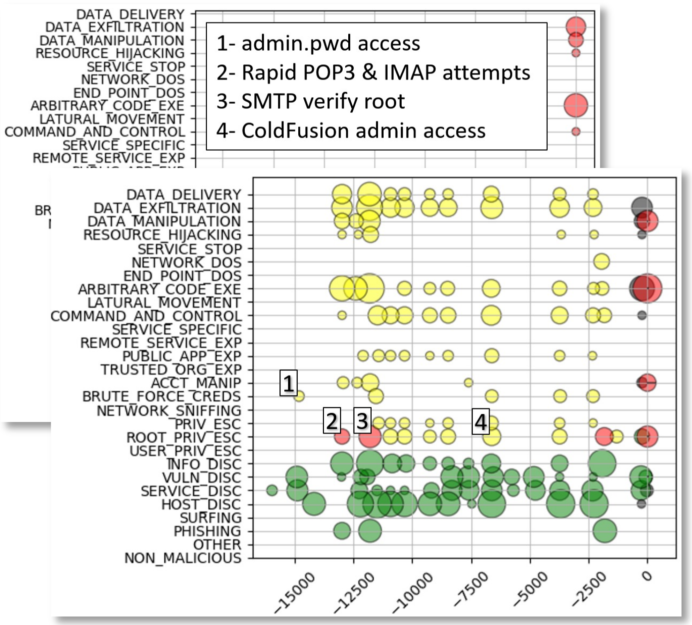
Our network-agnostic features allow HeAT to find similarities between strategies **regardless of adversary or network**

Thanks for listening!

Stephen Moskal
sfm5015@rit.edu
Linked-in: https://www.linkedin.com/in/stephen-moskal/
Music: https://www.mixcloud.com/shadw_moses/

HeATed Attack Campaign Part Two – HeAT Entropy Gain

HAC: HeATed Attack Campaigns



We needed a metric to aid the user in finding HAC's describing a diverse set of attack types and sufficiently capture the domain knowledge defined by the analyst

X: Attack Stage
Y: Predicted HeAT Value

h: HAC
d: dataset under test
t: training data

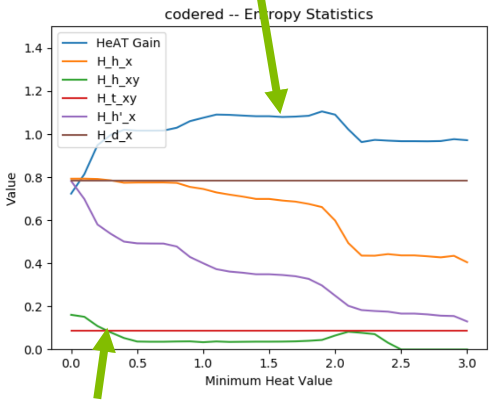
$$_{HAC}(X,Y) = H_h(X) + \underbrace{\left(H_d(X) - H_h'(X)\right)}_{\text{HAC Uniqueness from overall dataset}} - \underbrace{abs\left(H_h(X|Y) - H_t(X|Y)\right)}_{\text{Domain knowledge deviation adjustment}}$$

AIS Entropy of HAC

HAC Uniqueness from overall dataset

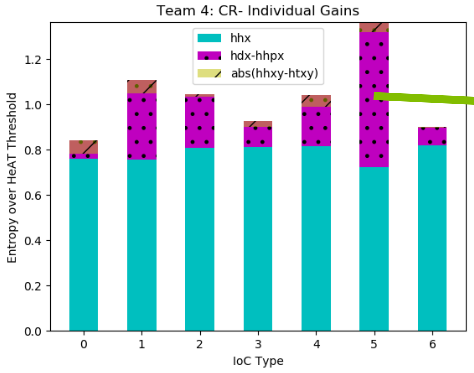
Domain knowledge deviation adjustment

Our network-agnostic features allowed HeAT to find similarities between strategies **regardless of adversary or network**

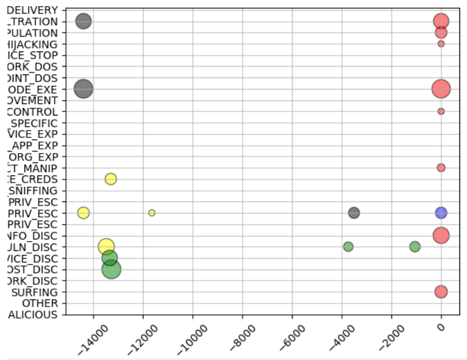


Optimal domain knowledge captured

HAC comparisons



HAC comparisons



PATRL: (*Pseudo Active TRansfer Learning*) to interpret cryptic alerts

"ET EXPLOIT Possible CVE-2014-3704 Drupal SQLi attempt URLENCODE1"

What type of attack is this describing??

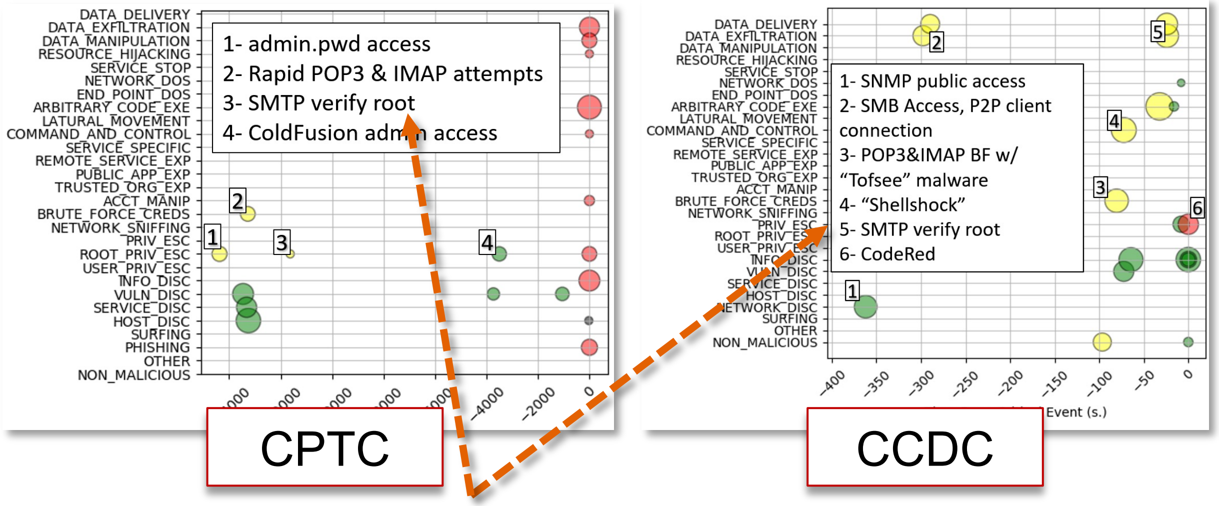
Approaches – **Transfer Learning** (ULMFiT), **Monte Carlo Dropout Uncertainty** (MCDU), **Pseudo-Active Transfer Learning**

Transfer LM w/ Text Source(s)	Top 1 Acc.	Top 3 Acc.
Multinomial Naive Bayes (No LM)	.5452	.8025
LM: Wikipedia (Default)	.3535	.61
LM: Wiki + IMDB	.4357	.75
LM: Wiki + MITRE ATT&CK	.5928	.8786
LM: Wiki + CPTC/CCDC Suricata	.6462	.9048
LM: Wiki + All Suricata (64k)	.6871	.85
LM: Wiki + CVE Database	.6975	.8929
LM: Wiki + All Cyber-relevant Texts	.8024	.9084
LM: Wiki + All Cyber + 1k Random PL's	**.8292**	**.98**

~1000 labeled signatures to classify 64k!

Heated Alert Triage (HeAT): Network Agnostic Extraction of Cyber Attack Campaigns

Approaches – **Alert Episode Heat** (captures the impact of initial triage), **Network-Agnostic Features**, **HeATed Attack Campaign** (w/ alert aggregation)

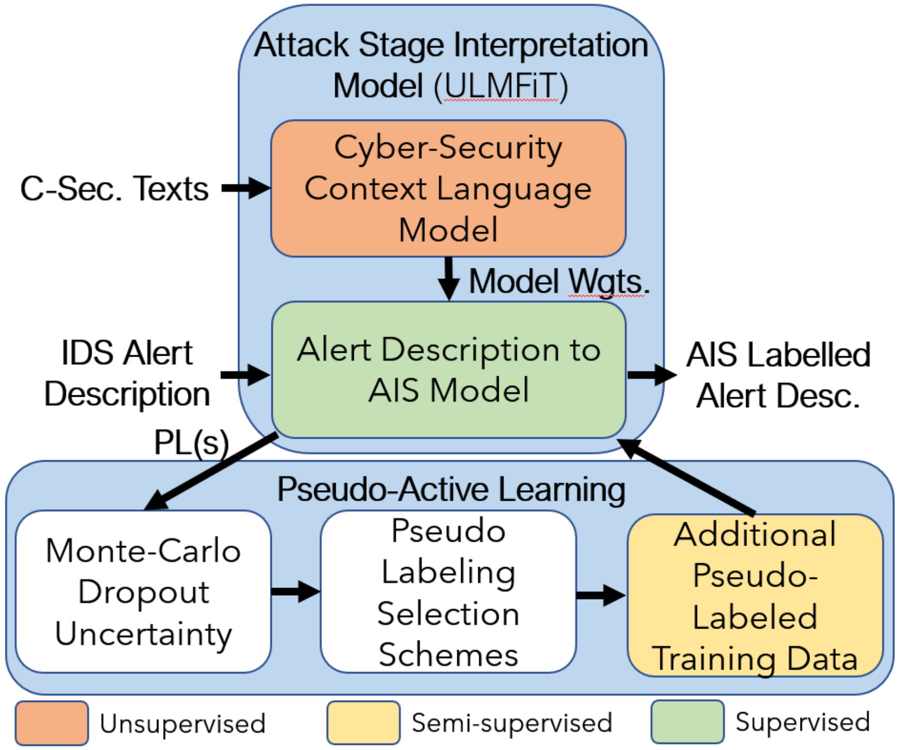


Our network-agnostic features allow HeAT to find similarities between strategies **regardless of adversary or network**

PATRL (Pseudo Active TRansfer Learning) to interpret cryptic alerts

- Problem: **how to translate cryptic alerts with limited expertise and time?**
  - SOC analysts may be only familiar with a small portion (~1%) of alerts – use AI/ML to help.
  - e.g., "ET EXPLOIT Possible CVE-2014-3704 Drupal SQLi attempt URLENCODE1"
    - Web-Attack, Code-Exe or Priv-Esc? Only 2.5% Suricata has CVE numbers to search for.

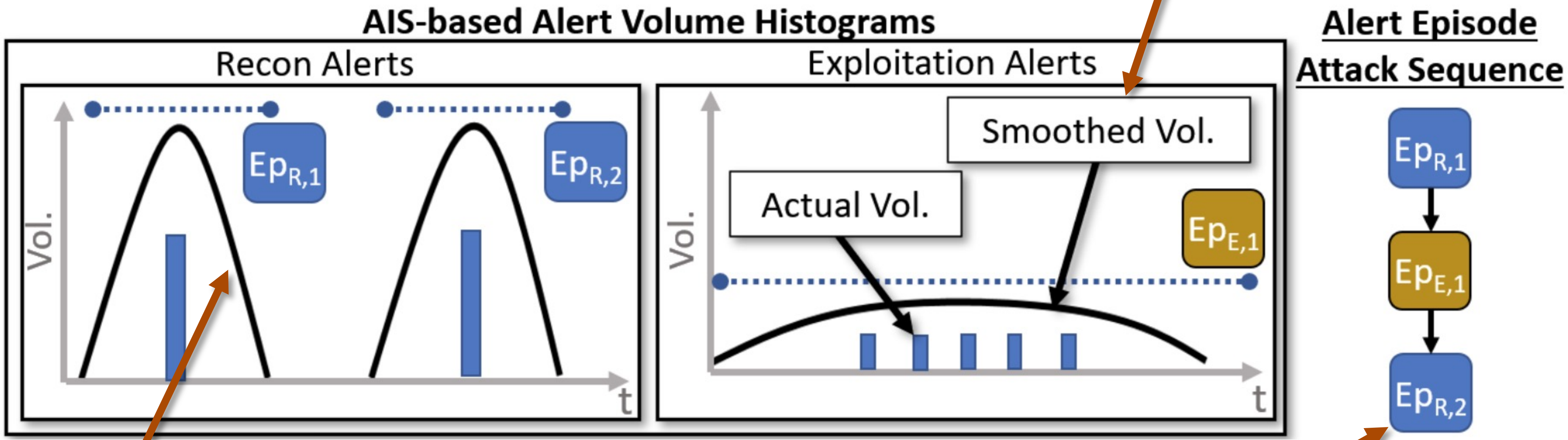
- **No existing works other than using SIEM & online info** to manually find the meaning of unknown alerts.
- Approach:
  - Use Transfer Learning to learn the cyber "language" and train an initial predictor w/ ~1% labeled data.
  - Use Monte-Carlo Dropout Uncertainty (MCDU) to measure the uncertainty of prediction.
  - Use Pseudo-Labeled (predicted) data based on MCDU to refine the prediction model.
  - Use MCDU to provide confidence in predicted labels.



Step 1: Dealing With High Volume of IDS Alerts – "Alert Episodes"

- **Problem**: IDS's produce an overwhelming amount of alerts per-day (~10k-1M)
  - Often many false positives or one 'action' causing many alerts (recon, scripting, etc.)
  - **Objective:** Consolidate similar alerts based on the attack type, IP addresses, and time

Method: Apply Gaussian Smoothing to the volumes of alerts for each source IP, for each attack stage

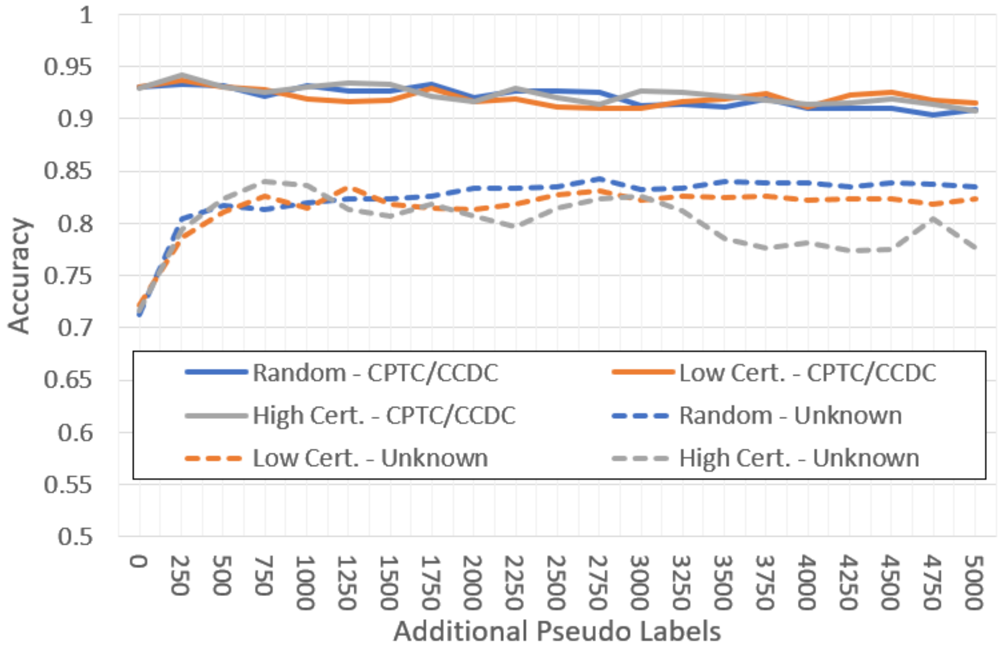


The 'decline' in alert volume signifies the end of an action

Episodes represent similar alerts that are likely to be caused by one action by an adversary

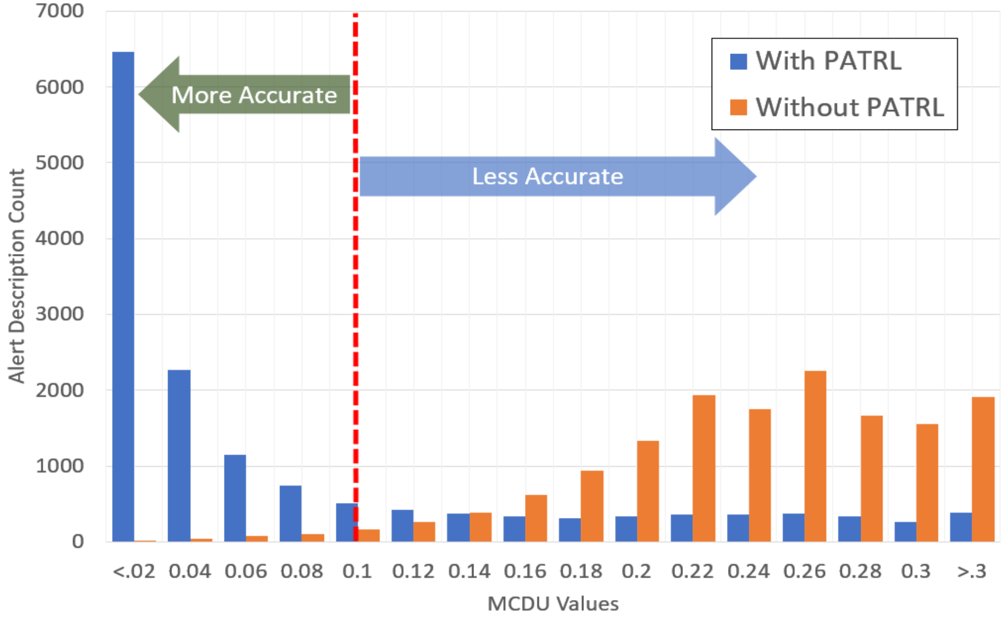
PATRL – performance analysis

- Transfer learning performs well with cyber-relevant text.
  - but suffers when used directly for unknown alerts.
- Iteratively adds in pseudo-labeled data improves pred. for unknown and maintains perf for the known ones.
- Users can use MCDU to differentiate the quality of prediction for unknown alerts.



Transfer LM w/ Text Source(s)	Top 1 Acc.	Top 3 Acc.
Multinomial Naive Bayes (No LM)	.5452	.8025
LM: Wikipedia (Default)	.3535	.61
LM: Wiki + IMDB	.4357	.75
LM: Wiki + MITRE ATT&CK	.5928	.8786
LM: Wiki + CPTC/CCDC Suricata	.6462	.9048
LM: Wiki + All Suricata (64k)	.6871	.85
LM: Wiki + CVE Database	.6975	.8929
LM: Wiki + All Cyber-relevant Texts	.8024	.9084
LM: Wiki + All Cyber + 1k Random PL's	**.8292**	**.98**

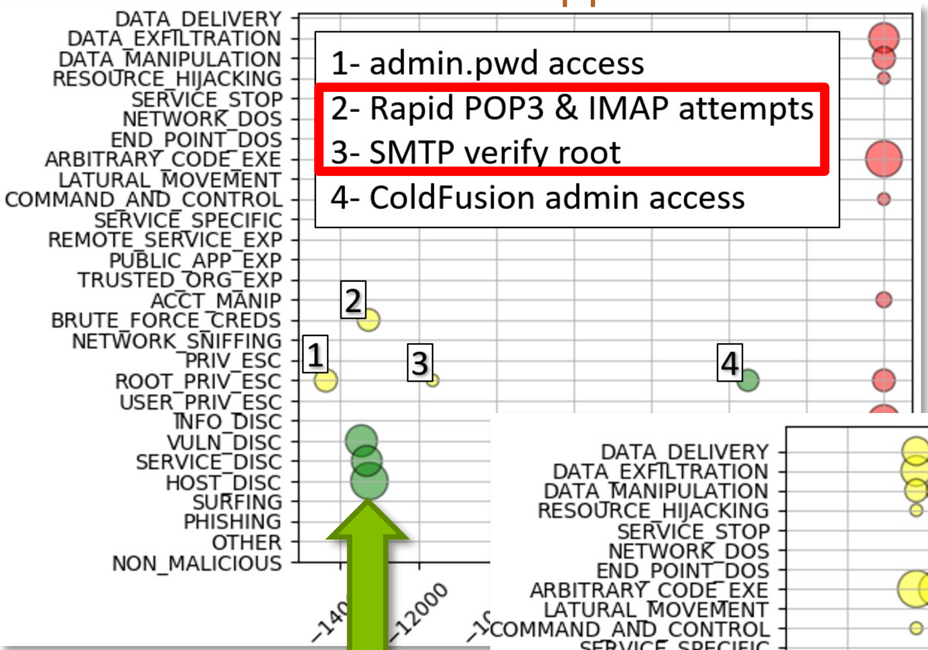
Training — Testing	CPTC/CCDC	Unknown Test
CPTC/CCDC	.9385 (.9742)	.7216 (.8001)
Unknown Test	.3116 (.62)	.9271 (.995)



HeATed Attack Campaign Examples – "CodeRed"

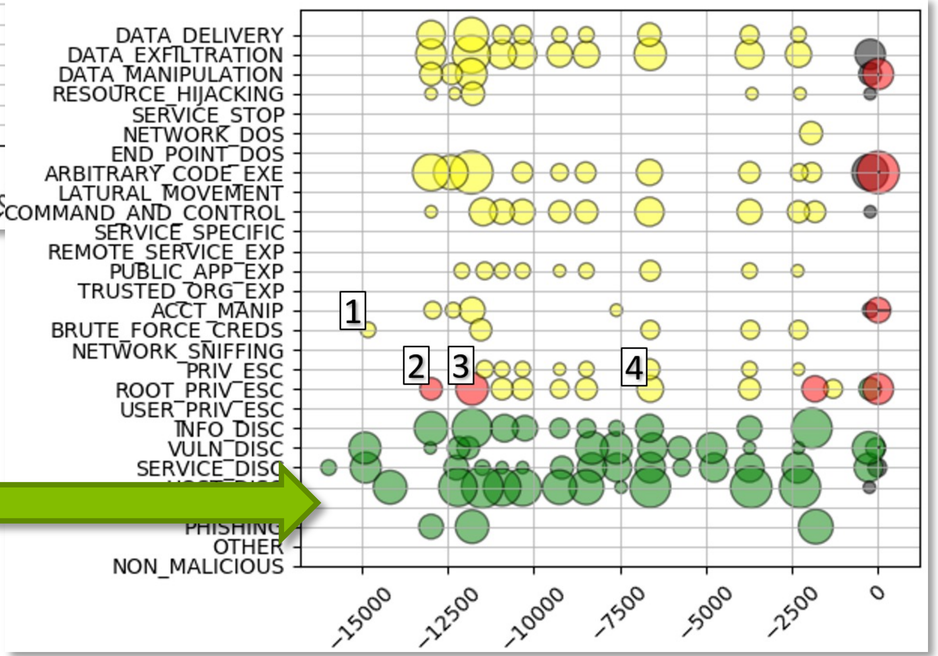
HeATing Different Adversaries (CPTC18)

"Calculated" Approach



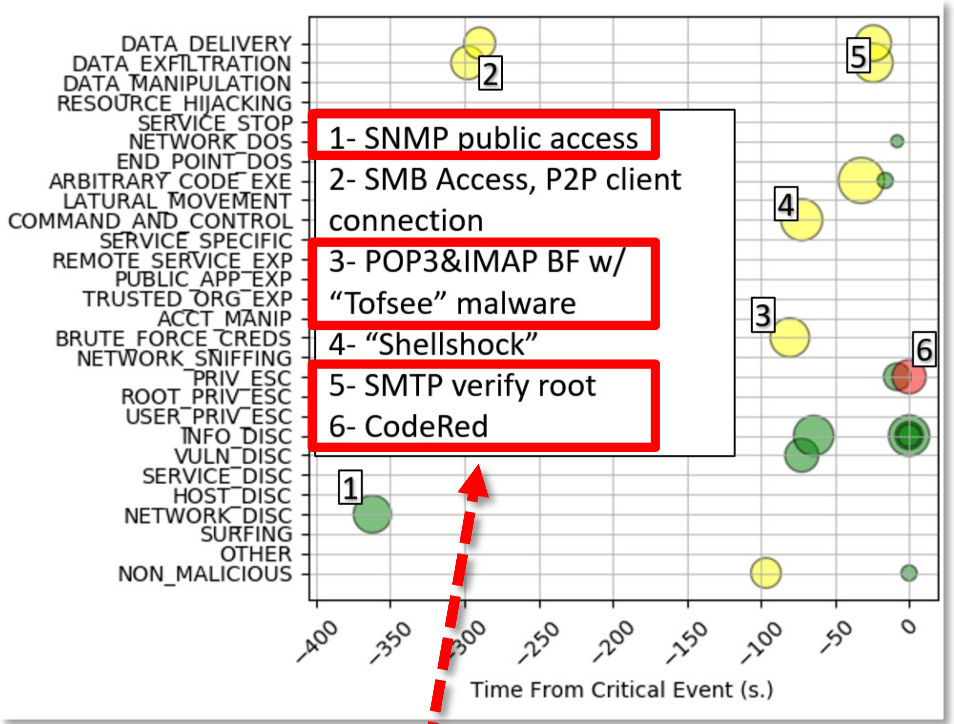
Same critical IoC & network, very different behaviors!

"The Script Kiddie"



10,000's of alerts represented as a handful of alert episodes

HeATing Different Networks (CCDC18 w/ CPTC observations)



Our network-agnostic features allow HeAT to find similarities between strategies **regardless of adversary or network**